

DISSERTAÇÃO DE MESTRADO

**ANÁLISE DA TRANSFORMADA
WAVELET DIRECIONAL ADAPTATIVA
NA CODIFICAÇÃO DE IMAGENS.**

Karen França de Oliveira

Brasília, janeiro de 2009

UNIVERSIDADE DE BRASÍLIA

FACULDADE DE TECNOLOGIA

**UNIVERSIDADE DE BRASÍLIA
FACULDADE DE TECNOLOGIA
DEPARTAMENTO DE ENGENHARIA ELÉTRICA**

**ANÁLISE DA TRANSFORMADA WAVELET DIRECIONAL
ADAPTATIVA NA CODIFICAÇÃO DE IMAGENS**

KAREN FRANÇA DE OLIVEIRA

ORIENTADOR: RICARDO LOPES DE QUEIROZ

DISSERTAÇÃO DE MESTRADO EM ENGENHARIA ELÉTRICA

PUBLICAÇÃO: PPGENE.DM - 363A/09

BRASÍLIA/DF: JANEIRO – 2009

UNIVERSIDADE DE BRASÍLIA
FACULDADE DE TECNOLOGIA
DEPARTAMENTO DE ENGENHARIA ELÉTRICA

ANÁLISE DA TRANSFORMADA WAVELET DIRECIONAL
ADAPTATIVA NA CODIFICAÇÃO DE IMAGENS

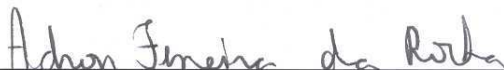
KAREN FRANÇA DE OLIVEIRA

DISSERTAÇÃO DE MESTRADO ACADÊMICO SUBMETIDA AO DEPARTAMENTO DE ENGENHARIA ELÉTRICA DA FACULDADE DE TECNOLOGIA DA UNIVERSIDADE DE BRASÍLIA, COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE MESTRE.

APROVADA POR:



RICARDO LOPES DE QUEIROZ, Dr, ENE/UNB
(ORIENTADOR)



ADSON FERREIRA DA ROCHA, Dr., ENE/UNB
(EXAMINADOR INTERNO)



EDUARDO ANTÔNIO BARROS DA SILVA, Dr., COPPE/UFRJ
(EXAMINADOR EXTERNO)

BRASÍLIA, 30 DE JANEIRO DE 2008.

FICHA CATALOGRÁFICA

OLIVEIRA, KAREN FRANÇA DE

Análise da transformada *wavelet* direcional adaptativa na codificação de imagens [Distrito Federal] 2009.

xiii, 117p., 210 × 297 mm (ENE/FT/UnB, Mestre, Telecomunicações, Processamento de Sinais, 2009). Dissertação de Mestrado. Universidade de Brasília. Faculdade de Tecnologia.

Departamento de Engenharia Elétrica.

1. Codificação de Imagem 2. Filtragem adaptativa

3. Transformadas *wavelet* 4. *Lifting* direcional

5. Número de condição

I. ENE/FT/UnB

II. Título (série)

REFERÊNCIA BIBLIOGRÁFICA

OLIVEIRA, K. F. (2009). Análise da transformada *wavelet* direcional adaptativa na codificação de imagens. Dissertação de Mestrado em Engenharia Elétrica com ênfase em Telecomunicações, Publicação PPGENE.DM - 363A/09, Departamento de Engenharia Elétrica, Universidade de Brasília, Brasília, DF, 117p.

CESSÃO DE DIREITOS

NOME DO AUTOR: Karen França de Oliveira.

TÍTULO DA DISSERTAÇÃO DE MESTRADO: Análise da transformada *wavelet* direcional adaptativa na codificação de imagens.

GRAU: Mestre ANO: 2009

É concedida à Universidade de Brasília permissão para reproduzir cópias desta dissertação de mestrado e para emprestar ou vender tais cópias somente para propósitos acadêmicos e científicos. O autor reserva outros direitos de publicação e nenhuma parte desta dissertação de mestrado pode ser reproduzida sem a autorização por escrito do autor.

Karen França de Oliveira

SQS 107 Bloco K Apto 504

70346-110 Brasília - DF - Brasil.

Dedicatória

Dedico esse trabalho a todos os familiares e amigos, que tiveram toda a compreensão nos momentos de ausência e que tudo fizeram para viabilizar a concentração de meus esforços nessa empreitada.

Karen França de Oliveira

Agradecimentos

Agradeço aos amigos que, diariamente, conviveram com a ânsia de que esse trabalho fosse finalizado, não só pela paciência que demonstraram, mas também, pela ajuda que ofertaram. Aos amigos que não participaram diretamente, igualmente gostaria de agradecer, pois além de aceitarem minhas sucessivas faltas ainda foram capazes de torcer para que tudo desse certo e, claro, para que terminasse logo. Por fim, agradeço imensamente à minha família, alvo mínimo de atenção nesse período, pela verdadeira dedicação a mim oferecida, permitindo o livre desembaraço de minhas atividades, e pelo contínuo incentivo que dela tenho recebido, essencial à transposição de qualquer limite.

Karen França de Oliveira

RESUMO

Esse trabalho apresenta um método de análise da estabilidade numérica de transformadas *wavelet* direcionais adaptativas via *lifting*. O objetivo é investigar a relação entre uma dada combinação de direções dos estágios de *lifting* e a qualidade de reconstrução das bases correspondentes. Esse novo método é baseado na variação em frequência do número de condição da matriz polifásica de síntese de bancos de filtros multidimensionais direcionais. As transformadas implementadas, caracterizadas por bancos de filtros dessa natureza, permitem adaptar localmente as direções de filtragem às direcionalidades de uma imagem, resultando em menores coeficientes de detalhe nas sub-bandas de alta frequência. Para tal, modificou-se o *kernel* da *wavelet* interpolante (6,6), inserindo-se informação direcional em seus estágios de *lifting*. Diferentes abordagens foram sugeridas para definir a combinação ótima de direções, no sentido de compressão. Os melhores resultados apontaram para escolha da direção de predição baseada em um algoritmo de minimização de distorção, adotando-se, em seguida, a mesma direção para o passo de atualização. Para realizar os testes de desempenho da transformada direcional implementada, substituiu-se os módulos de transformação dos codificadores JPEG2000 e SPIHT. Desse modo, efetivamente conseguiu-se maior compactação de energia para imagens com transições abruptas. Prosseguiu-se com a análise da estabilidade numérica das bases, fundamentada na variação do número de condição. Os procedimentos realizados mostraram que bases originadas por direções de predição distintas da de atualização realmente apresentam pior qualidade de reconstrução. Como o módulo de transformada é uma parte importante de um sistema de compressão de imagens, uma melhoria nesse módulo efetivamente é capaz de prover bons resultados em termos taxa-distorção para o sistema todo.

ABSTRACT

This work presents a method for the numerical stability analysis of adaptive lifting-based directional transforms. The goal is to investigate the relation between the directions set in the lifting scheme and the reconstruction quality of the corresponding bases. This new method is based on the variation in frequency of the condition number of the synthesis polyphase transfer matrix of multidimensional directional filter banks. The implemented transforms, characterized by such filters, allows for the adaptation of the filtering directions to match the local directionality of the image features, resulting in lower energy of the high-frequency subbands. In order to do so, the (6,6) interpolating wavelet kernel was modified, inserting directional information into the lifting steps. Different methods were suggested to determine the optimal set of directions, in terms of maximizing compression performance. Best results pointed to the prediction direction choice based on the distortion minimization algorithm, followed by the update step in the same direction. In order to test the directional transforms, the transform modules of JPEG2000 and SPIHT coders were replaced by the aforementioned transform algorithm. In this way, one can really achieve higher energy compaction for images with abrupt transitions. Next, we carried a numerical stability analysis based on the variation in frequency of the condition number of the transform. Results have shown that bases generated by using the prediction direction different from the update direction actually leads to worse reconstruction quality. As the transform is one of the crucial elements in transform coding systems, improvements in that module may effectively lead to overall coding improvements.

SUMÁRIO

| | | |
|----------|--|-----------|
| 1 | INTRODUÇÃO | 1 |
| 1.1 | CONTEXTUALIZAÇÃO..... | 1 |
| 1.2 | ORGANIZAÇÃO | 4 |
| 2 | COMPRESSÃO DE IMAGENS | 5 |
| 2.1 | CONCEITOS | 5 |
| 2.2 | COMPRESSÃO COM OU SEM PERDAS | 6 |
| 2.3 | EXPLORANDO A REDUNDÂNCIA | 8 |
| 2.3.1 | REDUNDÂNCIA ESTATÍSTICA..... | 8 |
| 2.3.2 | INFORMAÇÕES IRRELEVANTES..... | 9 |
| 2.3.3 | IRRELEVÂNCIA NA REPRESENTAÇÃO EM ESPAÇO DE CORES | 9 |
| 2.4 | ELEMENTOS DE UM SISTEMA DE COMPRESSÃO | 11 |
| 2.4.1 | CODIFICAÇÃO DE ENTROPIA..... | 11 |
| 2.4.2 | QUANTIZAÇÃO..... | 13 |
| 2.4.3 | TRANSFORMAÇÃO | 13 |
| 2.5 | MODELAGEM | 19 |
| 2.5.1 | TRANSFORMADAS SEPARÁVEIS | 20 |
| 3 | BANCOS DE FILTROS | 23 |
| 3.1 | BANCO DE FILTROS E RECONSTRUÇÃO PERFEITA | 23 |
| 3.2 | ANÁLISE DOS ERROS GERADOS EM UM BANCO QMF | 27 |
| 3.2.1 | ERROS DE <i>aliasing</i> E DE <i>imaging</i> | 27 |
| 3.2.2 | DISTORÇÃO DE AMPLITUDE E DE FASE..... | 30 |
| 3.2.3 | UM SISTEMA QMF SIMPLES COM CANCELAMENTO DE <i>alias</i> | 31 |
| 3.2.4 | REPRESENTAÇÃO POLIFÁSICA | 31 |
| 3.2.5 | LIMITAÇÕES IMPOSTAS PELA RESTRIÇÃO $H_1(z) = -H_0(z)$, EM UM BANCO QMF | 33 |
| 3.3 | BANCOS DE FILTROS COM M FAIXAS | 34 |
| 3.3.1 | EXPRESSÃO MATEMÁTICA DO SINAL RECONSTRUÍDO - M FAIXAS | 35 |
| 3.3.2 | REPRESENTAÇÃO POLIFÁSICA - M FAIXAS | 36 |
| 3.3.3 | RECONSTRUÇÃO PERFEITA - M FAIXAS | 38 |
| 3.4 | BANCOS DE FILTROS MULTIDIMENSIONAIS | 42 |

| | | |
|----------|--|------------|
| 3.4.1 | CONSIDERAÇÕES INICIAIS | 42 |
| 3.4.2 | CONCEITOS BÁSICOS SOBRE SINAIS MULTIDIMENSIONAIS..... | 43 |
| 3.4.3 | FILTROS DIGITAIS MULTIDIMENSIONAIS | 44 |
| 3.4.4 | AMOSTRAGEM DE SINAIS MULTIDIMENSIONAIS | 45 |
| 3.5 | FUNDAMENTOS DE SISTEMAS MULTITAXA MULTIDIMENSIONAIS..... | 48 |
| 3.5.1 | COMPONENTES BÁSICOS | 48 |
| 3.5.2 | REPRESENTAÇÃO POLIFÁSICA - MULTIDIMENSIONAL | 50 |
| 3.6 | TRANSFORMADAS <i>wavelet</i> E BANCOS DE FILTROS | 54 |
| 3.6.1 | HISTÓRICO | 54 |
| 3.6.2 | MODELAGEM | 55 |
| 4 | TRANSFORMADAS DIRECIONAIS IMPLEMENTADAS VIA <i>lifting</i> | 59 |
| 4.1 | HISTÓRICO | 59 |
| 4.2 | <i>Lifting</i> | 61 |
| 4.2.1 | MODELAGEM | 62 |
| 4.3 | <i>Lifting</i> BIDIMENSIONAL CONVENCIONAL | 67 |
| 4.4 | <i>Lifting</i> BIDIMENSIONAL DIRECIONAL | 69 |
| 4.4.1 | ESCOLHA ADAPTATIVA DAS DIREÇÕES | 72 |
| 4.4.2 | CORRELAÇÃO ENTRE AS DIREÇÕES DE FILTRAGEM | 73 |
| 5 | RESULTADOS | 75 |
| 5.1 | VALIDAÇÃO | 75 |
| 5.2 | IMPLEMENTAÇÃO DA TRANSFORMADA <i>wavelet</i> DIRECIONAL (D-DWT) | 75 |
| 5.2.1 | PRIMEIRA ABORDAGEM: DIREÇÃO DE ATUALIZAÇÃO IGUAL À DE PREDIÇÃO | 76 |
| 5.2.2 | SEGUNDA ABORDAGEM: NOVO CRITÉRIO DE SUAVIZAÇÃO PARA SE ESCOLHER A DIREÇÃO DE ATUALIZAÇÃO INDEPENDENTEMENTE..... | 77 |
| 5.3 | ANÁLISE DA ESTABILIDADE NUMÉRICA | 78 |
| 5.3.1 | METODOLOGIA | 80 |
| 6 | CONCLUSÕES | 95 |
| 6.1 | SUMÁRIO DO TRABALHO E AVALIAÇÃO DOS RESULTADOS | 95 |
| 6.2 | SUGESTÃO PARA TRABALHOS FUTUROS | 96 |
| | REFERÊNCIAS BIBLIOGRÁFICAS | 98 |
| | ANEXOS | 103 |
| I | MATRIZES COM AS VARIAÇÕES EM FREQUÊNCIA DO NÚMERO DE CONDIÇÃO E SUAS REPRESENTAÇÕES GRÁFICAS | 105 |

LISTA DE FIGURAS

| | | |
|------|--|----|
| 2.1 | Ilustração de uma imagem digitalizada e a interpretação das coordenadas das amostras | 6 |
| 2.2 | Componentes nos espaços de cores RGB e YC_bC_r para a imagem Lena | 10 |
| 2.3 | Visão global de um sistema de compressão..... | 11 |
| 2.4 | Evidenciação dos elementos de um sistema de compressão | 12 |
| 2.5 | Quantizador escalar simples com quatro símbolos de saída | 13 |
| 2.6 | Transformação do sinal original \mathbf{x} no sinal \mathbf{y} por simples somas e diferenças. . | 16 |
| 2.7 | Os sinais \mathbf{x}_c e \mathbf{y}_c muito se assemelham aos sinais \mathbf{x} e \mathbf{y} , quando as amostras pequenas foram ceifadas..... | 17 |
| 2.8 | Exemplo de transformação simples, considerando a média das amostras de um bloco | 18 |
| 2.9 | Implementação separável de uma transformação bidimensional, aplicada unidimensionalmente às linhas e, em seguida, às colunas..... | 21 |
| 2.10 | (a) das sub-bandas de uma transformada <i>wavelet</i> bidimensional: LL - simbolizando duas filtrações passa-baixa (do inglês, <i>Low-pass</i>); LH - simbolizando uma filtração passa-baixa (do inglês, <i>Low-pass</i>), seguida de uma filtração passa-alta (do inglês, <i>High-pass</i>); HL - simbolizando uma filtração passa-alta (do inglês, <i>High-pass</i>), seguida de uma filtração passa-baixa (do inglês, <i>Low-pass</i>); HH - simbolizando duas filtrações passa-alta (do inglês, <i>High-pass</i>). (b) Organização evidenciando os detalhes destacados em cada sub-banda: A - aproximação; V - detalhe vertical; H - detalhe horizontal; D - detalhe diagonal..... | 22 |
| 3.1 | Banco de Filtros Digitais: (a) banco de análise; (b) banco de síntese. | 24 |
| 3.2 | Banco de Filtros com M faixas criticamente decimado. | 24 |
| 3.3 | Operações básicas para a conversão da taxa de amostragem. O sobreamostrador ou expensor (a) insere $M - 1$ amostras nulas entre cada par de amostras originais, aumentando a taxa de amostragem por um fator M . O decimador (b) retém uma a cada M amostras do sinal original, reduzindo a taxa de amostragem por um fator M | 25 |
| 3.4 | (a) Banco de Filtros Espelhados em Quadratura; (b) repostas típicas de magnitude..... | 26 |

| | | |
|------|---|----|
| 3.5 | Duas possíveis repostas em magnitude para os filtros de análise: (a)sem sobreposição, e (b)com sobreposição. | 28 |
| 3.6 | Banco de Filtros com M faixas criticamente decimado | 34 |
| 3.7 | Representação polifásica de um banco de análise com M faixas..... | 37 |
| 3.8 | Representação polifásica de um banco de síntese com M faixas | 38 |
| 3.9 | Banco de Filtros Digitais: (a) Representação Polifásica de um banco de filtros com M faixas criticamente decimado; (b) Rearranjo a partir de identidades nobres; (c) Simplificação, em que $\mathbf{P}(z) = \mathbf{R}(z)\mathbf{E}(z)$ | 39 |
| 3.10 | Conjunto de pontos resultante da amostragem $x(n_0, n_1) = x_a(n_0 - n_1, n_0 + 2n_1)$ | 47 |
| 3.11 | Conjunto de amostras <i>quincunx</i> | 47 |
| 3.12 | Demonstração dos quatro componentes polifásicos gerados para amostragem do tipo hexagonal, localizados no interior do $FPD(\mathbf{M})$ | 52 |
| 3.13 | Representação em diagrama de blocos das relações entre os sinais $x(\mathbf{n})$, $x_k(\mathbf{n})$ e $y_k(\mathbf{n})$ | 52 |
| 3.14 | Decomposição de $x(\mathbf{n})$ em componentes polifásicos e sua recomposição. | 56 |
| 3.15 | Representação tempo-frequencial para (a)impulso, (b)senóide pura e (c) <i>wavelet</i> | 56 |
| 3.16 | Estrutura em árvore logarítmica que origina as <i>wavelets</i> | 57 |
| 3.17 | Cada estrutura <i>wavelet packet</i> origina um conjunto de bases, incluindo Walsh e Haar..... | 58 |
| 4.1 | Resultado da filtragem de um filtro adaptativo de Gabor da Imagem Bárbara. | 62 |
| 4.2 | Imagem Bárbara binarizada com vetores indicando as direções predominantes nos blocos, obtidas por meio de filtros de Gabor. | 63 |
| 4.3 | Representação em diagrama de blocos dos três passos de <i>lifting</i> : <i>Split</i> , <i>Prediction-P</i> e <i>Update-U</i> | 64 |
| 4.4 | Representação das direções de predição e de atualização de uma (2-D DWT) convencional separável. As circunferências sólidas não preenchidas representam \mathcal{G}_{00} , as tracejadas não preenchidas \mathcal{G}_{01} , as sólidas preenchidas \mathcal{G}_{10} e as tracejadas preenchidas \mathcal{G}_{11} . Em destaque encontram-se as amostras a serem preditas ou atualizadas. (a)Transformação entre linhas: Apenas amostras localizadas sob a mesma coluna poderão compor as funções de predição e de atualização.(b)Transformação entre colunas: Apenas amostras localizadas sob a mesma linha poderão compor as funções de predição e de atualização..... | 68 |
| 4.5 | Conjunto de direções propostas para a (2-D DWT) direcional adaptativa. as flechas sólidas representam os vetores \mathbf{d} e $-\mathbf{d}$, enquanto os tracejados representam $\tilde{\mathbf{d}}$ e $-\tilde{\mathbf{d}}$. As circunferências sólidas não preenchidas representam \mathcal{G}_{00} , as tracejadas não preenchidas \mathcal{G}_{01} , as sólidas preenchidas \mathcal{G}_{10} , e as tracejadas preenchidas \mathcal{G}_{11} | 70 |

| | | |
|------|---|----|
| 4.6 | Representação de métodos de implementação de 2-D DWT's direcionais adaptativas em que a direção é escolhida a partir da tangente de uma inclinação espacial θ_v , envolvendo, portanto, localizações sub- <i>pixel</i> . As posições inteiras são marcadas por “O”, as posições <i>half-pixel</i> são marcadas por “+”, enquanto as posições <i>quarter-pixel</i> são marcadas por “x”. | 72 |
| 4.7 | Esquemático da decomposição direcional em sub-bandas, considerando-se um nível completo de 2-D DWT. Os indicadores ao longo das setas referem-se às direções adotadas em cada passo de predição e de atualização realizado: DP1 - Direção de Predição 1; DU1 - Direção de <i>Update</i> 1; DLP2 - Direção de Predição da sub-banda <i>Low-pass</i> 2; DLU2 - Direção de <i>Update</i> da sub-banda <i>Low-pass</i> 2; DHP2 - Direção de Predição da sub-banda <i>High-pass</i> 2; DHU2 - Direção de <i>Update</i> da sub-banda <i>High-pass</i> 2. | 74 |
| 5.1 | Imagens de Teste: (a) Barbara; (b) Lena; (c) GoldHill; (d) Baboon..... | 83 |
| 5.2 | Curvas de $PSNR \times taxa$ resultantes da codificação da imagem Bárbara, utilizando-se a D-DWT com direções de predição e de atualização iguais..... | 84 |
| 5.3 | Curvas de $PSNR \times taxa$ resultantes da codificação da imagem Lena, utilizando-se a D-DWT com direções de predição e de atualização iguais..... | 85 |
| 5.4 | Curvas de $PSNR \times taxa$ resultantes da codificação da imagem Baboon, utilizando-se a D-DWT com direções de predição e de atualização iguais..... | 86 |
| 5.5 | Curvas de $PSNR \times taxa$ resultantes da codificação da imagem Goldhill, utilizando-se a D-DWT com direções de predição e de atualização iguais..... | 87 |
| 5.6 | Testes de suavização - imagem Bárbara: (a) com critério de suavização; (b) sem critério de suavização. | 88 |
| 5.7 | Testes de suavização - imagem Lena: (a) com critério de suavização; (b) sem critério de suavização. | 89 |
| 5.8 | Testes de suavização - imagem Baboon: (a) com critério de suavização; (b) sem critério de suavização. | 90 |
| 5.9 | Testes de suavização - imagem Goldhill: (a) com critério de suavização; (b) sem critério de suavização. | 91 |
| 5.10 | Esquemático da metodologia de composição da matriz polifásica de síntese a partir dos coeficientes da transformada inversa, considerando a aplicação à sub-banda LL como exemplo..... | 92 |
| 5.11 | Variação do número de condição, $\eta(\mathbf{E}(\omega_1, \omega_2))$, referente à <i>Wavelet</i> interpolante (6,6) convencional: $DP1 = 5$, $DU1 = 5$, $DLP2 = 5$, $DLU2 = 5$, $DHP2 = 5$ e $DHU2 = 5$. (a)Representação tridimensional, onde ω_1 e ω_2 são os eixos horizontais. (b)Representação bidimensional a cores da variação de $\eta(\mathbf{E}(\omega_1, \omega_2))$, contendo ω_1 e ω_2 nas linhas e colunas. | 93 |

| | | |
|------|--|-----|
| 5.12 | Variaco do nmero de condio, $\eta(\mathbf{E}(\omega_1, \omega_2))$, referente  combinao de direes $DP1 = 8$, $DU1 = 5$, $DLP2 = 5$, $DLU2 = 5$, $DHP2 = 5$ e $DHU2 = 5$, contendo, portanto, direo de predio diferente da de atualizao. (a)Representao tridimensional, onde ω_1 e ω_2 so os eixos horizontais. (b)Representao bidimensional a cores da variao de $\eta(\mathbf{E}(\omega_1, \omega_2))$, contendo ω_1 e ω_2 nas linhas e colunas. | 94 |
| I.1 | Variaco do nmero de condio, $\eta(\mathbf{E}(\omega_1, \omega_2))$, referente  <i>Wavelet</i> interpolante (6,6) convencional: $DP1 = 5$, $DU1 = 5$, $DLP2 = 5$, $DLU2 = 5$, $DHP2 = 5$ e $DHU2 = 5$ | 106 |
| I.2 | Variaco do nmero de condio, $\eta(\mathbf{E}(\omega_1, \omega_2))$, referente  combinao de direes $DP1 = 3$, $DU1 = 3$, $DLP2 = 3$, $DLU2 = 3$, $DHP2 = 3$ e $DHU2 = 3$ | 107 |
| I.3 | Variaco do nmero de condio, $\eta(\mathbf{E}(\omega_1, \omega_2))$, referente  combinao de direes $DP1 = 9$, $DU1 = 9$, $DLP2 = 5$, $DLU2 = 5$, $DHP2 = 5$ e $DHU2 = 5$ | 108 |
| I.4 | Variaco do nmero de condio, $\eta(\mathbf{E}(\omega_1, \omega_2))$, referente  combinao de direes $DP1 = 5$, $DU1 = 5$, $DLP2 = 9$, $DLU2 = 9$, $DHP2 = 5$ e $DHU2 = 5$ | 109 |
| I.5 | Variaco do nmero de condio, $\eta(\mathbf{E}(\omega_1, \omega_2))$, referente  combinao de direes $DP1 = 1$, $DU1 = 5$, $DLP2 = 5$, $DLU2 = 5$, $DHP2 = 5$ e $DHU2 = 5$ | 110 |
| I.6 | Variaco do nmero de condio, $\eta(\mathbf{E}(\omega_1, \omega_2))$, referente  combinao de direes $DP1 = 8$, $DU1 = 5$, $DLP2 = 5$, $DLU2 = 5$, $DHP2 = 5$ e $DHU2 = 5$ | 111 |
| I.7 | Variaco do nmero de condio, $\eta(\mathbf{E}(\omega_1, \omega_2))$, referente  combinao de direes $DP1 = 5$, $DU1 = 5$, $DLP2 = 1$, $DLU2 = 5$, $DHP2 = 5$ e $DHU2 = 5$ | 112 |
| I.8 | Variaco do nmero de condio, $\eta(\mathbf{E}(\omega_1, \omega_2))$, referente  combinao de direes $DP1 = 5$, $DU1 = 5$, $DLP2 = 2$, $DLU2 = 5$, $DHP2 = 5$ e $DHU2 = 5$ | 113 |
| I.9 | Variaco do nmero de condio, $\eta(\mathbf{E}(\omega_1, \omega_2))$, referente  combinao de direes $DP1 = 5$, $DU1 = 5$, $DLP2 = 4$, $DLU2 = 5$, $DHP2 = 5$ e $DHU2 = 5$ | 114 |
| I.10 | Variaco do nmero de condio, $\eta(\mathbf{E}(\omega_1, \omega_2))$, referente  combinao de direes $DP1 = 5$, $DU1 = 5$, $DLP2 = 6$, $DLU2 = 5$, $DHP2 = 5$ e $DHU2 = 5$ | 115 |
| I.11 | Variaco do nmero de condio, $\eta(\mathbf{E}(\omega_1, \omega_2))$, referente  combinao de direes $DP1 = 5$, $DU1 = 5$, $DLP2 = 7$, $DLU2 = 5$, $DHP2 = 5$ e $DHU2 = 5$ | 116 |

I.12 Variação do número de condição, $\eta(\mathbf{E}(\omega_1, \omega_2))$, referente à combinação de direções $DP1 = 5$, $DU1 = 5$, $DLP2 = 9$, $DLU2 = 5$, $DHP2 = 5$ e $DHU2 = 5$ 117

LISTA DE SÍMBOLOS

| | |
|-------|---|
| 1-D | <i>one-dimensional</i> |
| 2-D | <i>two-dimensional</i> |
| 3-D | <i>three-dimensional</i> |
| M-D | <i>multi-dimensional</i> |
| CCIR | <i>Consultative Committee for International Radio</i> |
| CD | <i>Compact Disc</i> |
| DCT | <i>Discrete Cousine Transform</i> |
| DWT | <i>Discrete Wavelet Transform</i> |
| D-DWT | <i>Directional Discrete Wavelet Transform</i> |
| DFB | <i>Directional Filter Bank</i> |
| FIR | <i>Finite Impulse Response</i> |
| FPD | <i>Fundamental Parallelepiped</i> |
| GB | <i>Gigabytes</i> |
| IEC | <i>International Electrotechnical Commission</i> |
| ISO | <i>International Organisation for Standardisation</i> |
| ITU | <i>International Telecommunication Union</i> |
| JPEG | <i>Joint Photographic Experts Group</i> |
| MPEG | <i>Motion Picture Experts Group</i> |
| MSE | <i>Mean Square Error</i> |
| PSNR | <i>Peak Signal to Noise Ratio</i> |
| QMF | <i>Quadrature Mirror Filters</i> |
| RD | <i>Rate-Distortion</i> |
| STFT | <i>Short Time Fourier Transform</i> |
| SPIHT | <i>Set Partitioning in Hierarchical Trees</i> |

Capítulo 1

Introdução

1.1 Contextualização

Nas últimas décadas temos assistido uma enorme transformação no que se refere à própria maneira de nos comunicarmos e, a cada dia, mais e mais rápidas mudanças são verificadas. Essa transformação inclui a crescente utilização da rede mundial de computadores (*Internet*), bem como a variedade de aplicativos que dela se utilizam; o desenvolvimento explosivo das comunicações móveis; além do crescimento cada vez maior da importância das vídeo-comunicações. É importante também notar que, nesse novo cenário que se forma, todos esses meios de comunicação se inter-relacionam. Por exemplo, freqüentemente um usuário deseja descarregar um vídeo via *internet* e assistir em seu telefone celular. Uma aplicação complexa como essa, do ponto de vista da engenharia, passou a integrar o conjunto de necessidades básicas de um usuário que, à época da implantação das comunicações móveis, apenas desejava realizar e receber ligações. Nesse contexto, é simples notar a tendência de crescimento das demandas dos usuários por mais e mais recursos, o que está intimamente ligado ao interesse do mercado em satisfazê-las, desenvolvendo tecnologias que as suportem. Por trás de toda essa revolução multimídia estão as técnicas de compressão de dados, que são essenciais para sua viabilização. O advento da TV digital com todas as funcionalidades que dela se espera, como, por exemplo, a utilização de um canal de retorno para que o próprio usuário possa mandar as informações que desejar, seria impossível sem compressão. Com essa evolução, compressão de dados deixou de ser um assunto afeto apenas a um grupo seleto de engenheiros e cientistas para se tornar ubíquo.

Pode-se, então, esclarecer um pouco mais o que é compressão e, principalmente, para que se precisa dela. Provavelmente, muitos já devem ter ouvido falar de JPEG (*Joint Photographic Experts Group*) [1] e MPEG (*Motion Picture Experts Group*) [2], que são, respectivamente, padrões utilizados na representação de imagem e de vídeo. Em padrões como esses, são utilizados algoritmos de compressão de dados para reduzir o número de

bits necessários à representação de uma imagem ou de uma seqüência de vídeo a uma dada qualidade. Assim, compressão de dados pode ser entendida como a ciência ou, até mesmo, a arte de representar uma informação de maneira mais compacta. Essa representação mais compacta é obtida identificando-se e utilizando-se características da própria estrutura existente nos dados. Dados podem ser caracteres em um arquivo texto, números que representam amplitudes de um sinal de áudio, de uma imagem ou de uma seqüência de vídeo, ou seqüências de números gerados por qualquer outro processo.

A razão pela qual necessitamos de compressão é que cada vez mais geramos e utilizamos informações em formato digital, isto é, na forma de números representados por *bytes* de dados, e a quantidade de dados envolvidos pode ser muito grande. Por exemplo, para que se possa representar digitalmente um segundo de vídeo sem compressão (usando-se o formato CCIR 601 ou qualidade de vídeo *Broadcast*), necessita-se de mais que 20 MB (*megabytes*). Isso significaria que um disco DVD-5 (*Digital Video Disc*), que possui capacidade de 4,7 GB (*gigabytes*), poderia guardar apenas 3,7 minutos de vídeo sem compressão. Se considerarmos que a duração média de um filme é de 120 minutos, se torna fácil perceber porque precisamos de compressão. Para representarmos 2 minutos de música sem compressão e em qualidade de CD (*Compact Disc*), com taxa de 44.100 amostras por segundo e 16 *bits* por amostra, seriam necessários mais de 84 milhões de *bits* para representá-la. Assim, descarregar uma música de um sítio da Internet demoraria muito tempo, pelo menos muito mais tempo do que usuários típicos desejariam esperar.

Surge, então, um questionamento pertinente: dado o crescimento explosivo da quantidade de dados que precisamos transmitir ou armazenar, por que então não se coloca o foco no desenvolvimento de melhores tecnologias de transmissão e de armazenagem? Isso também tem acontecido, mas não tem sido suficiente. Vários avanços significativos permitem que um volume cada vez maior de informação possa ser armazenado ou transmitido, incluindo a mais nova mídia para armazenamento de dados de alta densidade chamada BD (*Blu-ray Disc*). A capacidade dessa nova mídia varia de 25 GB (camada simples) a 50 GB (camada dupla), ou seja, em apenas um disco podem ser armazenados filmes *Full HD* (resolução de 1920×1080 pixels), de até 4 horas de duração, sem perdas perceptíveis. Para que possamos ter idéia de quão grande foi o salto dessa nova tecnologia em relação à sua antecessora, a mídia DVD, compararemos as capacidades de armazenamento. Os DVDs possuem a capacidade de armazenar 4,7 GB de dados, enquanto que um CD armazena em média de 700 MB. Já os chamados DVDs de camada dupla (*dual-layer*), podem armazenar até 8,5 GB. Para o novo disco óptico BD, o salto foi muito maior, visto que sua capacidade varia de 25GB (camada simples) a 50GB (camada dupla). No entanto, é importante observar que, ainda hoje, a mídia BD sucessora ainda não se popularizou no Brasil, dado que a própria mídia e seus reprodutores (*Blu-ray players*), são ainda muito caros. Por fim, deve-se também notar que existem situações em que a capacidade de transmissão e de armazenagem não é passível de ser significativamente aumentada, o que dá ainda maior destaque às técnicas de compressão. Um exemplo bem simples de aplicação com essa restrição é a que envolve

transmissão por radiodifusão, já que a quantidade de informação que pode ser transmitida será sempre limitada pelas características da atmosfera.

Sendo assim, mesmo com os esforços efetivos destinados a aumentar a capacidade dos meios de transmissão e de armazenagem de dados, as técnicas de compressão possuem papel fundamental para que novos serviços possam ser ofertados aos usuários ou, até mesmo, para baixar os custos desses serviços e, conseqüentemente, expandir o acesso. É por essas razões que várias empresas da área de tecnologia realizam grandes investimentos na pesquisa de compressão de dados, majoritariamente de vídeo e de imagem, buscando atender demandas relacionadas a entretenimento.

Como será visto no Capítulo 2, um módulo importante de um sistema de compressão de imagens é o que realiza uma transformação nos dados, isto é, um módulo capaz de representar os mesmos dados em um formato diferente, porém com maior concentração de energia em uma quantidade menor de símbolos. Dessa maneira, pode-se, até mesmo, optar por descartar alguns símbolos para se economizar na quantidade total de *bits*, incorrendo em menor perda de informação do que se dispensássemos alguns dos dados originais. Em outras palavras, a transformação dos dados originais permite que se incorra em uma diminuição da taxa necessária para representá-los acompanhada de uma redução desproporcionalmente menor na qualidade do sinal recuperado. Existem muitas ferramentas capazes de representar um sinal de maneira mais compacta, favorecendo a sua compressão. Entre elas, destacam-se a transformada de Fourier e a transformada discreta de cossenos. No entanto, também será discutido que o grau de compactação observado nos elementos transformados por uma dada ferramenta depende das características do próprio sinal. Essa sentença expõe uma informação importante: não se pode dizer que exista um tipo de transformada ótima para qualquer sinal. Como consequência, vê-se que é importante buscar uma transformada que possa ser adaptada ao sinal de interesse.

Nesse trabalho, será dado enfoque a uma ferramenta que se tornou muito popular nos últimos anos: as transformadas *wavelet*. Convencionalmente, uma transformada *wavelet* aplicada a imagens, portanto, bidimensional, é implementada de forma separável, resultando da aplicação em cascata de duas transformadas *wavelet* unidimensionais. Será visto que esse tipo de implementação não é o mais adequado para representar sinais com direcionalidades localizadas. Usualmente, imagens naturais contêm informações direcionais significantes, que podem ser aproximadas por bordas lineares em nível local. Essas bordas podem não ser horizontais ou verticais, e se isso não for considerado, serão obtidos coeficientes de alta freqüência com maior amplitude nos dados transformados. Dado que quanto maior a compactação de energia, maior a possibilidade de compressão de uma imagem, nesse caso restará prejudicada a capacidade de compressão. Busca-se, então, uma maneira de se rotacionar as bases de representação, alinhando-as com as direções predominantes de uma imagem. Sabe-se que uma transformada *wavelet* pode ser implementada de forma eficiente pelo método denominado *lifting scheme* [3],[4]. Nesse método, cada filtro FIR da *wavelet*

pode ser fatorado em diversos passos de *lifting*, denominados separação (do inglês, *split*), predição (do inglês, *prediction*) e atualização (do inglês, *update*). Dessa maneira, pode-se inserir a informação direcional em cada um dos passos, promovendo uma representação adaptativa ao conteúdo da imagem.

Considerando a necessidade em se obter uma representação compacta de uma imagem, o presente projeto tem a função de evidenciar a escolha adequada dos parâmetros direcionais afetos a transformadas *wavelet* bidimensionais não-separáveis cujas direções de filtragem sejam adaptáveis a direcionalidades localizadas, provendo melhoria na capacidade de compressão desses sinais. Para esse fim, desenvolveu-se um novo método de análise da estabilidade numérica dessas transformadas, baseada na variação em frequência do número de condição da matriz polifásica de síntese do banco de filtros correspondente.

1.2 Organização

O presente trabalho é organizado da seguinte maneira: o segundo capítulo discute conceitos introdutórios em compressão de imagens, enumerando os principais componentes de um sistema dessa natureza e suas funcionalidades. Nesse capítulo, é dado enfoque no módulo de transformação de imagens, dada a sua fundamental importância para o trabalho. No capítulo 3 é feito um estudo sobre banco de filtros, desde a modelagem de sistemas unidimensionais com M bandas até o caso multidimensional. No fim desse capítulo é feita a associação de transformadas *wavelet* com uma estrutura especial de banco de filtros. Já no capítulo 4 é feita uma análise sobre transformadas direcionais, destacando o método de fatoração *lifting*. Nessa oportunidade é feita uma importante discussão sobre a problemática da escolha das direções de cada estágio de *lifting*. O capítulo 5 mostra os resultados obtidos na implementação da transformada direcional, além de descrever uma nova abordagem para a avaliação de sua estabilidade numérica, fundamentada na variação em frequência do número de condição de matrizes polifásicas. Por fim, no sexto capítulo é feito um sumário do trabalho, juntamente com uma avaliação crítica dos resultados obtidos, terminando com sugestões para trabalhos futuros.

Capítulo 2

Compressão de Imagens

2.1 Conceitos

Compressão é a arte ou a ciência de se representar dados com a menor quantidade de informação possível a uma qualidade aceitável, aproveitando-se para isso das próprias características dos dados ou, até mesmo, dos usuários da informação. Assim, compressão de imagens nada mais é do que a aplicação das técnicas de compressão a imagens digitais.

Uma imagem pode ser definida como uma função bidimensional $f(x, y)$, em que x e y são coordenadas espaciais, e o valor da função em cada ponto coordenado é chamado intensidade ou nível de cinza da imagem. Quando as coordenadas x e y assumem finitos valores, ou seja, quantidades discretas, diz-se que a imagem é digital. Logo, pode-se notar que uma imagem digital é composta por um número finito de elementos, que possuem uma localização particular em um plano e um valor associado. Estes elementos são conhecidos como *pixels* (do inglês, *picture element*).

Para os propósitos de processamento e de compressão de imagens, apenas imagens digitais devem ser consideradas, já que é pressuposto destas áreas a possibilidade de tratamento computacional dos dados. O processo de digitalização das coordenadas é conhecido como amostragem, enquanto a da amplitude, como quantização. Assim, para os propósitos desta tese, uma imagem será considerada uma sequência de valores amostrados:

$$x[n_1, n_2], \quad 0 \leq n_1 < N_1, \quad 0 \leq n_2 < N_2 \quad (2.1)$$

A primeira coordenada n_1 é entendida como índice para as linhas da imagem, enquanto n_2 , como índice para as colunas. Isto é ilustrado na Figura 2.1.

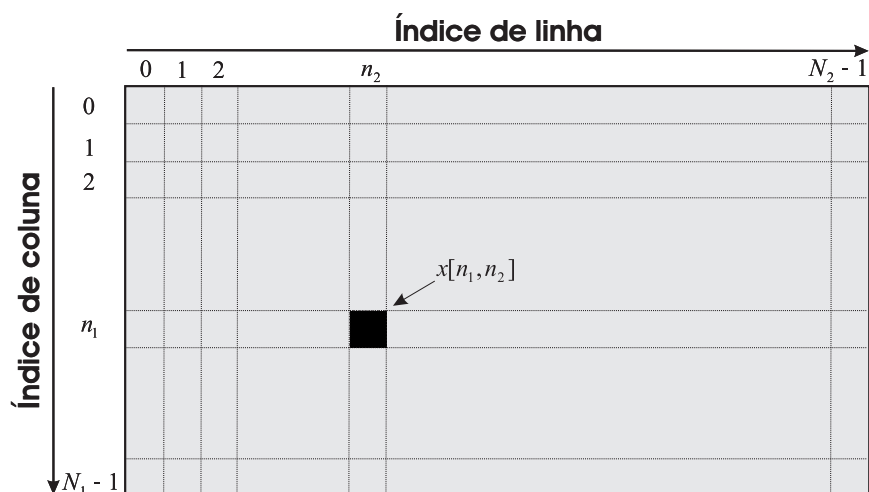


Figura 2.1: Ilustração de uma imagem digitalizada e a interpretação das coordenadas das amostras

2.2 Compressão com ou sem perdas

A principal questão a ser respondida em compressão de sinais é como se pode representar o sinal de interesse com a menor quantidade possível de *bits*, incorrendo em uma perda de informação aceitável. Considerando que o sinal original tenha sido digitalizado com B *bits* por amostra, todos estes seriam, em princípio, necessários à reconstrução perfeita do sinal. No entanto, pode-se considerar a estatística das amostras, e se obter uma representação sem perdas com menos *bits* por amostra, baseada, por exemplo, na correlação entre amostras vizinhas. Em se tratando especificamente de imagens, alguma perda de informação é normalmente aceita por três razões principais:

- a Perdas significantes podem ser toleradas pelo sistema visual humano sem prejudicar o entendimento das cenas.
- b Na maioria dos casos, a imagem digitalizada já é uma representação imperfeita do mundo real.
- c Métodos de compressão sem perdas normalmente não são suficientes para se atingir os requisitos de altas taxas de compressão demandada pelas aplicações.

Apesar de, na maioria das vezes, ser recomendado o uso de compressão com perdas, em algumas aplicações perdas são inaceitáveis. Na área médica, por exemplo, uma pequena perda de informação é capaz de ocultar ou inserir artefatos na imagem, suficientes para invalidar o diagnóstico do especialista. A compressão sem perdas também deve ser utilizada em aplicações em que a imagem será repetidamente editada e recomprimida, de tal sorte que a acumulação dos erros de compressão com perdas se tornaria inaceitável. Note que

apenas se torna vantajoso aceitar a introdução de pequenos erros se uma representação mais eficiente do sinal de interesse puder ser obtida, isto é, se menos *bits* forem gastos. Assim, quanto maior a distorção aceita, espera-se que menor seja quantidade de *bits* necessários na representação, ou seja, maior a compressão da informação. Definido o limiar de perda aceitável, deve-se então procurar a representação que incorra na menor quantidade de *bits*. Para analisar a solução de compromisso entre taxa e distorção, deve-se obter uma medida adequada desta.

As medidas de distorção podem ser divididas em dois grupos: subjetivas e objetivas. A medida é dita subjetiva quando a qualidade é avaliada por pessoas. No entanto, essas observações são necessariamente parciais, prejudicando a objetividade, e também apresentam certa instabilidade, já que os sentimentos de cada pessoa influenciam no processo. Além disso, a definição da distorção está ligada à aplicação, não sendo, na maioria dos casos, tão menor ou aceitável quanto as pessoas possam achar.

No que se refere a medidas objetivas, existem diferentes maneiras de se calcular a distorção por uma função predefinida $\mathbf{D}(\mathbf{x}, \hat{\mathbf{x}})$, entre a imagem original $\mathbf{x}[n_1, n_2]$, e a imagem reconstruída $\hat{\mathbf{x}}[n_1, n_2]$. A medida mais utilizada é o erro quadrático médio (MSE - do inglês "Mean Squared Error"), definido por:

$$MSE \triangleq \frac{1}{N_1 N_2} \sum_{n_1=0}^{N_1-1} \sum_{n_2=0}^{N_2-1} (\mathbf{x}(n_1, n_2) - \hat{\mathbf{x}}(n_1, n_2))^2 \quad (2.2)$$

Na área de processamento de imagens, o MSE é normalmente utilizado no cálculo de uma grandeza conhecida como PSNR (do inglês, "Peak-to-peak Signal to Noise Ratio"), definida por:

$$PSNR \triangleq 10 \log_{10} \frac{(2^B - 1)^2}{MSE} \quad (2.3)$$

A PSNR é expressa em dB (decibels). Imagens reconstruídas com qualidade geralmente apresentam PSNR em torno de 30 dB. Esta medida é bastante difundida na literatura, mas, infelizmente, nem sempre coincide com a opinião de um especialista. O olho humano, por exemplo, não observa pequenas variações de intensidade entre pixels vizinhos, sendo, no entanto, sensível à variações no valor médio e ao contraste em regiões maiores. Uma outra deficiência desta função é a de considerar apenas medidas locais, ou seja, diferença entre pixels, não capturando artefatos globais como efeito de bloco, "ringing", entre outras degradações estruturais.

2.3 Explorando a redundância

Sem qualquer compressão, a imagem da Figura 2.1, por exemplo, pode ser representada com $N_1 N_2 B$ bits. Porém, caso a imagem apresente informações redundantes, torna-se viável representar a mesma informação com um número menor de bits. Isso é compressão. Nesta seção discutiremos brevemente a natureza da redundância para motivar a o entendimento de um sistema completo de compressão de imagens, que será apresentado na próxima seção.

2.3.1 Redundância Estatística

Considere dois inteiros x_1 e $x_2 \in [0, 1, \dots, 2^B - 1]$, correspondendo a duas amostras adjacentes de uma imagem. Sem compressão, os dois inteiros poderiam ser representados por $2B$ bits. Suponha, no entanto, que o descompressor sabe *a priori* que a imagem é binária, ou seja, só apresenta valores 0 ou 1. Claramente, é suficiente indicar apenas um bit, obtendo uma compressão de $B : 1$. Suponha ainda, que também é informação *a priori* que os dois valores são sempre iguais. Neste caso, obtém-se uma compressão ainda maior de $2B : 1$. Na maioria dos casos, o descompressor não terá todo o conhecimento prévio descrito. Mais comum, no entanto, é saber que um conjunto de valores apresenta maior probabilidade de ocorrência que outros. Por exemplo, sabendo que existe uma probabilidade muito alta que $x_1 \in \{0, 1\}$, espera-se gastar pouco mais de 1 bit para indicar o seu valor atual. Como veremos, espera-se gastar tantos bits quanto $H(X_1)$, em que X_1 é uma variável aleatória que representa o conhecimento *a priori* do descompressor a respeito dos valores assumidos de x_1 e $H(X_1)$ é a função de distribuição estatística de X_1 , conhecida como sua entropia. Uma suposição razoável é a de que x_1 e x_2 sejam iguais, pois representam pixels vizinhos e, provavelmente, a imagem deva ser suave e, portanto, mudanças bruscas não são muito comuns. Neste caso, não se deve gastar muitos bits para indicar x_2 , dado que x_1 já foi identificado na seqüência de bits comprimida. Uma importante forma de se obter uma informação *a priori* é se considerar, por exemplo, a estatística conjunta das variáveis X_1 e X_2 , que correspondem aos valores assumidos, e o número de bits necessários para representar x_2 , dado que x_1 é conhecido, pode ser dado pela entropia condicional $H(X_2|X_1)$. Por hora, é suficiente notar que o número médio de bits utilizados na representação sem erro de uma imagem depende de suas propriedades estatísticas, isto é, quanto maior a correlação entre os pixels, menor deve ser a quantidade de bits gastos para representá-los. No pior caso, em que nenhuma redundância é observada, todos os valores são equiprováveis e independentes, nenhuma compressão é possível, e os $N_1 N_2 B$ bits deverão ser usados. No outro extremo, quando um pequeno conjunto de valores for suficiente para prever os demais, podem ser obtidas elevadas taxas de compressão.

2.3.2 Informações Irrelevantes

O aproveitamento da redundância entre as amostras descrito é uma das maneiras capazes de permitir uma representação exata da informação, porém com um número reduzido de *bits*. No entanto, em muitos casos, parte da informação das amostras é irrelevante, escolhendo-se uma representação inexata, porém bem menos custosa. Os dois exemplos ilustram situações em que isto se verifica.

- Irrelevância visual: Se a densidade de amostras ultrapassar a resolução perceptível pelo usuário humano, o excesso de informação é irrelevante.
- Irrelevâncias para uma aplicação específica: Em algumas aplicações, especialmente médicas ou militares, existem partes da imagem que deve ter uma resolução muito maior que outras. No primeiro caso, caso o especialista deseje analisar uma parte de um órgão, por exemplo, todos os outros componentes do quadro são irrelevantes, e este excesso de informação deve ser eliminado.

Uma maneira de explorar esta irrelevância é transformar as amostras originais em um novo conjunto de valores capaz de capturar a informação em poucos *bits*. No caso mais simples, a transformação pode apenas envolver sub-amostragem ou o descarte das partes menos relevantes. Pode-se ainda explorar a redundância das amostras restantes para se obter maior compressão, como a característica de suavidade já mencionada. Mais detalhes sobre transformações serão vistos no Capítulo 4.

2.3.3 Irrelevância na representação em espaço de cores

Um exemplo interessante de irrelevância ocorre na representação a cores de imagens. O observador humano apresenta menor sensibilidade a variações de cor e saturação do que a mudanças na intensidade. Para os propósitos de compressão de imagens, esta propriedade é usualmente aproveitada mapeando-se a imagem originalmente representada no espaço de cores *RGB* em um outro espaço de luminâncias e crominâncias YC_bC_r , utilizando-se uma popular transformação linear usada em padrões como JPEG e JPEG2000 [5], [6]. Esta conversão entre esses espaços de cor transformação é dada por:

$$\begin{bmatrix} x_Y \\ x_{C_b} \\ x_{C_r} \end{bmatrix} = \begin{bmatrix} 0,299 & 0,587 & 0,114 \\ -0,169 & -0,0331 & 0,5 \\ 0,5 & -0,419 & -0,0813 \end{bmatrix} \begin{bmatrix} x_R \\ x_G \\ x_B \end{bmatrix} \quad (2.4)$$

Pode-se então notar que o primeiro componente de crominância x_{C_b} é uma versão escalonada da diferença entre a original componente azul (B - do inglês "Blue") e a luminância (Y), mais especificamente:

$$x_{Cb} = 0,564(x_B - x_Y) \quad (2.5)$$

De maneira similar, a outra crominância x_{Cr} é obtida da diferença entre o vermelho (R - do inglês "Red"):

$$x_{Cr} = 0,713(x_R - x_Y) \quad (2.6)$$

A Figura 2.2 ilustra as componentes no espaço RGB, bem como as componentes do espaço YC_bC_r para a imagem Lena.

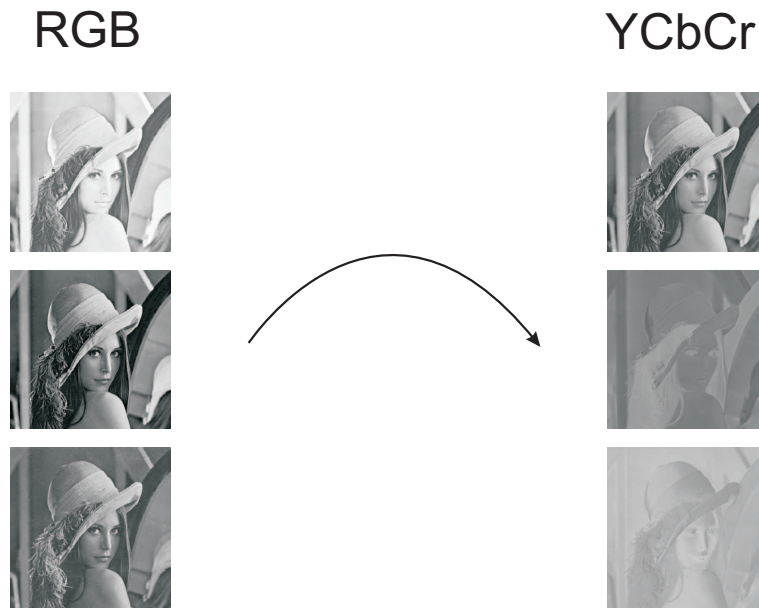


Figura 2.2: Componentes nos espaços de cores RGB e YC_bC_r para a imagem Lena

É comum, portanto, modelar a reduzida sensibilidade do sistema visual humano a variações rápidas nas cores reduzindo a resolução das crominâncias. Mais detalhadamente, costuma-se trabalhar com a representação no espaço YC_bC_r e ainda sub-amostrar as crominâncias por um fator 2 em cada uma das direções, horizontal e vertical. Neste sentido, as informações consideradas irrelevantes para a representação a cores de uma imagem, considerando a falta de sensibilidade do usuário da informação, teriam sido eliminadas.

Por fim, não se pode esquecer a principal questão a ser considerada em um sistema de compressão com perdas: a de minimizar a distorção para uma dada taxa de *bits* ou, equivalentemente, mimizar a taxa para uma dada distorção. Uma medida adequada de distorção deverá, portanto, refletir a informação relevante. Mais precisamente, se as imagens original \mathbf{x} e reconstruída $\hat{\mathbf{x}}$ diferirem apenas em aspectos irrelevantes, então $D(\mathbf{x}, \hat{\mathbf{x}})$ deverá ser nula.

2.4 Elementos de um sistema de compressão

A Figura 2.3 mostra a concatenação dos sistemas de compressão e de descompressão em termos de dois mapeamentos, M_1 e M_2^{-1} , respectivamente. Para que a compressão seja sem perdas, exige-se que $M_2^{-1} = M_1^{-1}$. Existem, porém, casos em que M_1 não é inversível, caso de compressão com perdas, em que será utilizada a notação M_2^{-1} para reforçar a idéia de que o sistema descompressor é apenas uma aproximação de inversa do sistema compressor. Uma maneira alternativa de entender o sistema compressor é como uma enorme tabela de referência (*lookup table*) com $2^{N_1 N_2 B}$ entradas, já que, em princípio, toda a vizinhança deverá ser incluída no contexto.

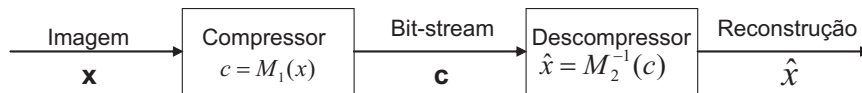


Figura 2.3: Visão global de um sistema de compressão

Pode-se, então, detalhar cada um destes sistemas em seus componentes. Como mostrado na Figura 2.4, o primeiro passo é o que transforma as amostras originais componentes da imagem em um novo conjunto, com características mais adequadas à compressão. Formalizando-se este passo, tem-se: $\mathbf{y} = T(\mathbf{x})$, em que $\mathbf{y} \equiv \{y(k_1, k_2)\}$ é uma outra seqüência bidimensional e finita, tendo $K_1 K_2$ elementos. As propriedades do operador T serão discutidas no Capítulo 4, por ora basta ressaltar que, normalmente, esta operação é inversível. Assim, o descompressor é capaz de utilizar uma transformada inversa, T^{-1} , sem que qualquer distorção seja inserida nesta etapa. O segundo passo é o que permite representar, de uma maneira aproximada, a seqüência de amostras transformadas por uma seqüência de índices quantizados. Assim, obtém-se $\mathbf{q} = Q\mathbf{y}$, e $\mathbf{q} \equiv q(p_1, p_2)$ representa uma seqüência bidimensional e finita com $P_1 P_2$ elementos. Geralmente, a quantidade de amostras $P_1 P_2$ não é maior e é, provavelmente, até menor que o número de amostras transformadas $K_1 K_2$. Essa etapa de quantização, sim, insere distorção no sistema de tal forma que o descompressor utiliza apenas uma inversa aproximada Q_2^{-1} . Finalmente, os índices de quantização são codificados entropicamente para formar o *bit-stream* final, que será transmitido ou armazenado, por exemplo. Escreve-se $\mathbf{c} = C(\mathbf{q})$. Este passo é inversível, não acrescentando distorção no processo, e o descompressor pode, então, recuperar os índices de quantização por $\mathbf{q} = C^{-1}(\mathbf{c})$. Veremos com um pouco mais de detalhe os passos de Codificação, Quantização e Transformação.

2.4.1 Codificação de entropia

O propósito da codificação de entropia é o de explorar a redundância estatística dos índices de quantização $q(p_1, p_2)$. As amostras transformadas e, por conseqüência, as

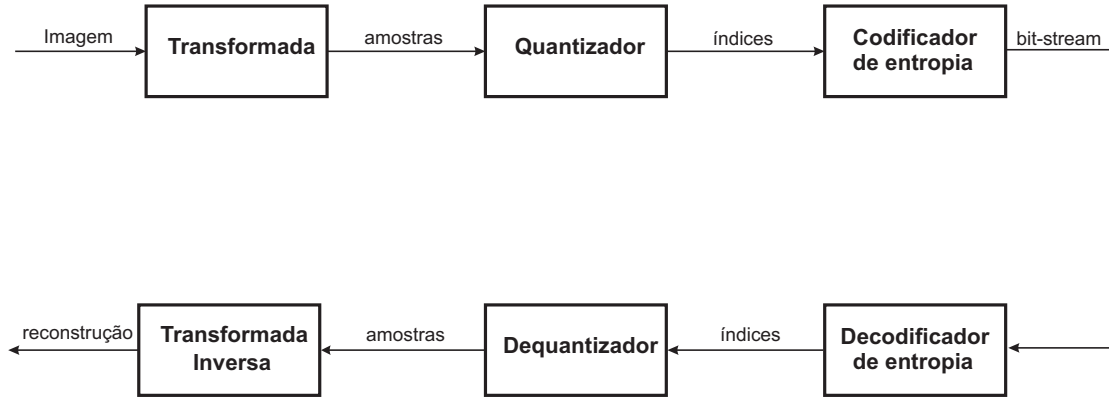


Figura 2.4: Evidenciação dos elementos de um sistema de compressão

quantizadas foram concebidas de forma a garantir que esta redundância seja espacialmente localizada. Dessa maneira, espera-se que a maior parte da energia esteja concentrada nos coeficientes de baixa frequência. Idealmente, as variáveis $Q(p_1, p_2)$ deveriam ser estatisticamente independentes, isto é, os passos de transformação e quantização deveriam ser suficientes para aproveitar toda a redundância da seqüência, decorrelacionando totalmente as amostras. Como um simples exemplo, considere que existam quatro índices de quantização possíveis, com suas probabilidades de ocorrência associadas:

$$P(Q = q) = \begin{cases} 1/2 & \text{se } q = 0 \\ 1/4 & \text{se } q = 1 \\ 1/8 & \text{se } q = 2 \\ 1/8 & \text{se } q = 3 \end{cases} \quad (2.7)$$

A partir das estatísticas, pode-se então determinar um código ótimo, isto é, capaz de prover uma representação média com menor número de *bits*.

$$\begin{array}{ll} q = 0 & \text{"1"} \\ q = 1 & \text{"01"} \\ q = 2 & \text{"001"} \\ q = 3 & \text{"000"} \end{array} \quad \rightarrow \quad (2.8)$$

O número médio de *bits* gastos na codificação de cada amostra seria:

$$\frac{1}{2} \cdot 1 + \frac{1}{4} \cdot 2 + \frac{1}{8} \cdot 3 + \frac{1}{8} \cdot 3 = 1\frac{3}{4} \text{ bits}. \quad (2.9)$$

Nota-se que o número de *bits* gasto com a codificação ótima é menor do que se apenas gastássemos 2 *bits* para indicar cada uma das quatro amostras. Normalmente, em aplicações práticas de compressão de imagens, pode-se ainda obter maior redução na taxa de *bits*. Vê-se, portanto, que a etapa de transformação realmente reduz a redundância dos dados,

permitindo que a etapa de quantização reduza bastante a quantidade de *bits* a ser usada na representação.

2.4.2 Quantização

A quantização, por si só, é responsável por introduzir distorção no sistema. Logo, em sistemas de compressão sem perdas, não pode existir quantização. No caso mais simples, pode-se mapear cada amostra transformada, $y(k_1, k_2)$, independentemente, em um índice de quantização correspondente, $q(k_1, k_2)$. Este processo é conhecido como quantização escalar. A quantização escalar associa cada índice de quantização com um intervalo no eixo real, de acordo com:

$$q(k_1, k_2) = i, \text{ se } y(k_1, k_2) \in I_i,$$

em que os intervalos, I_i , são disjuntos e cobrem todo o eixo real. A Figura 2.5, por exemplo, sugere uma divisão em quatro intervalos. Nesse caso, ter-se-ão quatro índices correspondentes. Conclui-se então que a inversa aproximada da quantização deve, a partir do índice $q(k_1, k_2)$, fornecer alguma representação pertencente ao intervalo correspondente, $\hat{y}(k_1, k_2)$. Uma maneira bem imediata é atribuir a $\hat{y}(k_1, k_2)$ o ponto médio do intervalo $I_{q(k_1, k_2)}$.

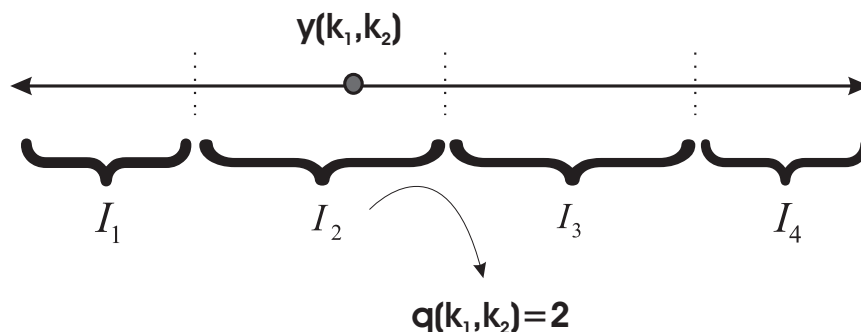


Figura 2.5: Quantizador escalar simples com quatro símbolos de saída

2.4.3 Transformação

A transformada é responsável por manipular as amostras originais da imagem em uma forma que simplifique as operações de codificação e de quantização. Por outro lado, a transformada deve capturar a dependência existente entre as amostras originais, para que as amostras transformadas $y(k_1, k_2)$ e os índices de quantização $q(k_1, k_2)$ exibam dependências fortemente localizadas. Para tal, a transformada deve ser capaz de separar as

informações relevantes das não-relevantes, para que as irrelevantes possam ser identificadas e mais fortemente quantizadas ou, até mesmo, descartadas. Felizmente, é possível construir transformadas capazes de atingir esses objetivos, pelo menos parcialmente. Idealmente, como já mencionado, se desejaria que as amostras transformadas se tornassem estatisticamente independentes.

Apenas para motivar o entendimento dos conceitos apresentados, será mostrado um simples exemplo. Mais precisamente, considere que os componentes $x(0)$, $x(1)$, $x(2)$ e $x(3)$ de um sinal serão substituídos por outros quatro números, $y(0)$, $y(1)$, $y(2)$ e $y(3)$, que serão combinações lineares dos elementos de $\{x\}$. Por exemplo, as combinações poderão ser somas e diferenças:

$$\begin{aligned} y(0) &= x(0) + x(1) & y(2) &= x(2) + x(3) \\ y(1) &= x(0) - x(1) & y(3) &= x(2) - x(3) \end{aligned} \quad (2.10)$$

Será, então, analisado como poderão ser recuperados os componentes originais do sinal e qual o propósito dessa nova representação. Primeiramente, nota-se que a transformação pode ser invertida utilizando-se também somas e subtrações, além da multiplicação por um fator de escala:

$$\begin{aligned} x(0) &= 0,5(y(0) + y(1)) & x(2) &= 0,5(y(2) + y(3)) \\ x(1) &= 0,5(y(0) - y(1)) & x(3) &= 0,5(y(2) - y(3)) \end{aligned} \quad (2.11)$$

Adiantando-se um pouco o assunto, as transformadas podem ser classificadas em três categorias:

- (a) Transformadas sem perdas (ortogonais): matrizes de transformação ortogonais e unitárias
- (b) Transformadas inversíveis (biortogonais): matrizes de transformação inversíveis
- (c) Transformadas com perdas: matrizes de transformação não inversíveis

Uma transformada unitária sem perdas pode ser representada por uma matriz de rotação, como em 2.12, que mostra a chamada matriz de rotação de Givens. Portanto, o sinal transformado possui o mesmo comprimento que o sinal original. Em outras palavras, o sinal original simplesmente será representado em novos eixos perpendiculares. Isto se aplica, por exemplo, à Transformada de Fourier e suas versões puramente reais, como a DCT (do inglês, *Discrete Cosine Transform*).

$$\mathbf{M} = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix} \quad (2.12)$$

Diferentemente, nas transformações biortogonais, podem ser alterados comprimentos e ângulos. Assim, os novos eixos não serão necessariamente perpendiculares, mas ainda assim nenhuma informação é perdida, ou seja, pode-se ainda obter o sinal original a partir dos coeficientes transformados. Um passo irreversível, por exemplo, seria o de eliminar os componentes de baixa amplitude, passo conhecido na área de compressão de sinais como quantização, já descrito. Desse modo, já se pode inferir que também as chamadas transformadas *wavelet* ortogonais poderão ser representadas por matrizes ortogonais e unitárias, enquanto as biortogonais poderão ser representadas por matrizes inversíveis, conservando, ainda, a característica de reconstrução perfeita.

Propõe-se adotar uma representação matricial do exemplo apresentado. Considerando-se \mathbf{x} e \mathbf{y} como vetores, obtém-se:

$$\begin{bmatrix} y(0) \\ y(1) \\ y(2) \\ y(3) \end{bmatrix} = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & -1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & -1 \end{bmatrix} \begin{bmatrix} x(0) \\ x(1) \\ x(2) \\ x(3) \end{bmatrix} \quad (2.13)$$

A matriz que recupera o vetor original, neste caso, é a de transformação dividida por dois:

$$\begin{bmatrix} x(0) \\ x(1) \\ x(2) \\ x(3) \end{bmatrix} = 1/2 \begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & -1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & -1 \end{bmatrix} \begin{bmatrix} y(0) \\ y(1) \\ y(2) \\ y(3) \end{bmatrix} \quad (2.14)$$

É importante observar que, ao se multiplicar as matrizes de transformação direta e inversa, obtém-se uma matriz identidade 4×4 , o que confirma o fato de a transformada ser inversível. Note que ao dividir a matriz de transformação direta por $\sqrt{2}$, as linhas desta matriz serão vetores unitários, ou seja, a matriz deixará de ser biortogonal para se tornar ortogonal.

Considerando-se então a transformada de um sinal como uma nova representação desse, essa nova representação somente será útil se, nos coeficientes transformados, for possível identificar padrões que não seriam evidentes nas amostras originais. Isto significa que o novo padrão deve ser mais bem aproveitado pelos algoritmos computacionais que, em regra, possuem melhor desempenho quando o sinal apresenta amplitudes grandes e baixas em seqüência. Considere, por exemplo, a seguinte seqüência \mathbf{x} de números:

$$x(0) = 1,2 \quad x(1) = 1,0 \quad x(2) = -1,0 \quad x(3) = -1,2.$$

Pode-se então obter sua representação \mathbf{y} , computando-se as somas e as diferenças entre as amostras:

$$y(0) = 2,2 \quad y(1) = 0,2 \quad y(2) = -2,2 \quad y(3) = 0,2.$$

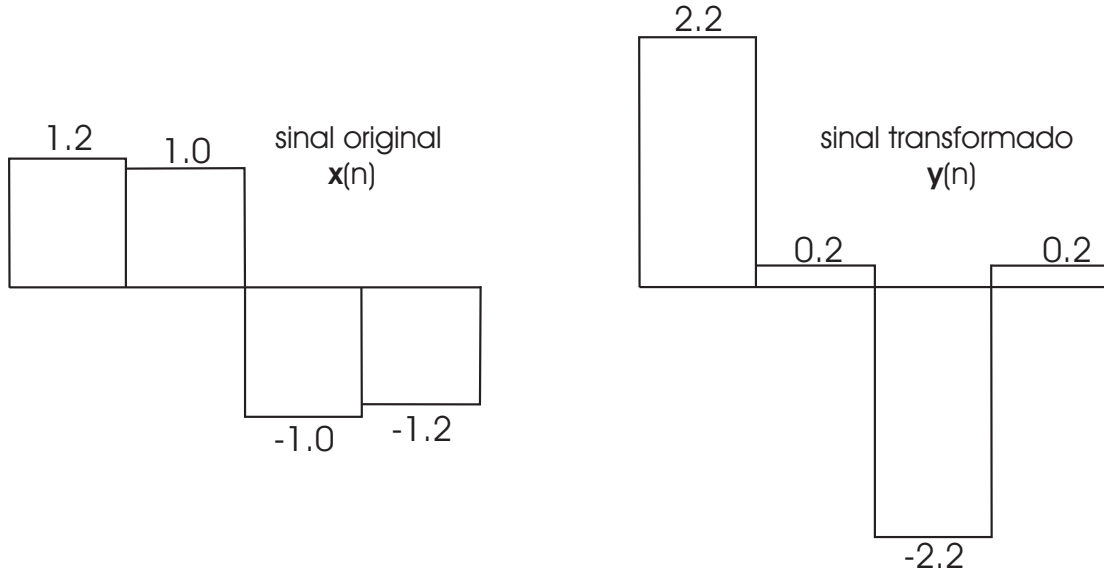


Figura 2.6: Transformação do sinal original \mathbf{x} no sinal \mathbf{y} por simples somas e diferenças.

Dada a nova representação, podem-se notar algumas peculiaridades do novo sinal. As diferenças $y(1)$ e $y(3)$ são muito maiores que as somas $y(2)$ e $y(4)$, ou seja, as amostras 2 e 4 foram praticamente ceifadas. É verdade que, para se reconstruir perfeitamente o sinal original, necessita-se das quatro amostras. Porém, considere uma versão comprimida \mathbf{y}_c do sinal \mathbf{y} , em que as amostras abaixo do limiar 0,2 foram ceifadas:

$$y_c(0) = 2,2 \quad y_c(1) = 0 \quad y_c(2) = -2,2 \quad y_c(3) = 0.$$

O sinal reconstruído a partir dessa versão comprimida então seria:

$$x_c(0) = 1,1 \quad x_c(1) = 1,1 \quad x_c(2) = -1,1 \quad x_c(3) = 1,1.$$

A partir do simples exemplo apresentado, vê-se que pode ser bastante desejado trocar a reconstrução perfeita em favor da eliminação de amostras transformadas e conseqüente redução da taxa de informação. Assim, torna-se importante utilizar transformações capazes

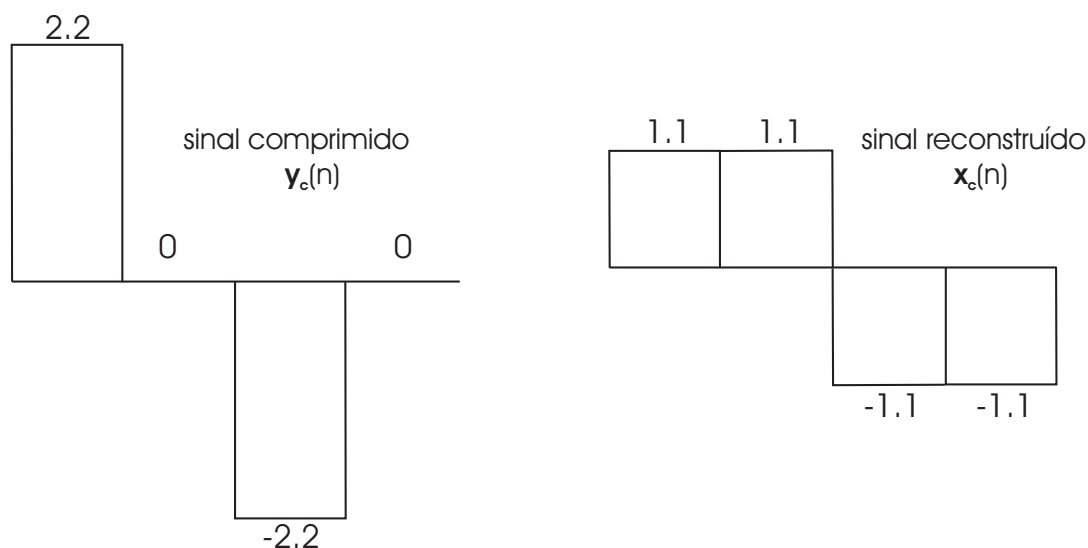


Figura 2.7: Os sinais x_c e y_c muito se assemelham aos sinais x e y , quando as amostras pequenas foram ceifadas.

de concentrar ao máximo a informação em poucas amostras, podendo-se então eliminar as demais, incorrendo-se em menor perda do que se ceifássemos amostras do sinal original. Observe que o sinal reconstruído no exemplo ainda apresenta características semelhantes às do sinal original.

Desde já, pode-se, portanto, observar que não existe uma única transformada ótima para todos os sinais. Logo, deve-se adequar o tipo de transformada às peculiaridades do sinal de interesse. Alguns tipos de sinais já possuem peculiaridades reconhecidas. Um sinal de voz, por exemplo, tem variações irregulares exigindo, portanto, intervalos não uniformes de amostragem. Já as imagens, que são sinais de comprimento finito, requerem tratamento especial para reduzir efeito de bloco. Neste caso, extensões simétricas combinam com filtros também simétricos.

Isto posto, pode-se novamente identificar a questão central em compressão de sinais: representar sinais típicos com o menor número de funções convenientes, obtendo-se a menor taxa de informação possível para uma dada qualidade ou obtendo-se a máxima qualidade para uma dada taxa. Bases tradicionais, como Bessel, Legendre e Fourier, decorrem de equações diferenciais. Já as *wavelet* não. Uma razão evidente é que essas equações não incluem dilatações.

É importante observar que, na maioria das aplicações, o sinal a ser transformado é ilimitado ou, pelo menos, excede o tamanho da transformada. Como conseqüência, a informação deverá ser processada em blocos e a escolha pela utilização de transformadas finitas implica no processo de divisão em blocos, como já mencionado na seção anterior. Sinais bidimensionais como, por exemplo, imagens, são usualmente particionados em blocos quadrados com $m \times m$ amostras cada. Assim, as transformadas bidimensionais de bloco

representam cada bloco da imagem como uma combinação de “blocos protótipos”, $s_{q_1, q_2}(\mathbf{n})$. Nesse processo, como cada um dos blocos é transformado independentemente, é de se esperar que efeitos de bloco possam ser observados na imagem reconstruída.

Apenas para motivar o entendimento da observação feita será mostrado um exemplo. Dividindo-se uma imagem em blocos de 2×2 *pixels*, cada um dos blocos pode ser representado pela média das amostras que o compõe e as diferenças entre as três outras amostras originais do bloco e a média, isto é:

$$\begin{cases} \frac{1}{4} \sum_{i,j \in \{0,1\}} \mathbf{x}(m+i, n+j), & \text{para } m,n \text{ pares} \\ \mathbf{x}(m, n) - \mathbf{y}(2 \lfloor \frac{m}{2} \rfloor, 2 \lfloor \frac{n}{2} \rfloor) & , \text{ para os demais casos} \end{cases} \quad (2.19)$$

Esta transformação é ilustrada na Figura 2.8

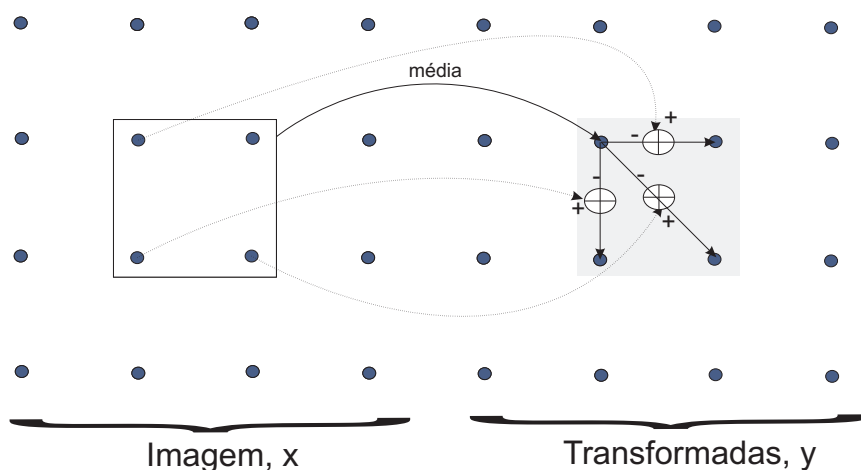


Figura 2.8: Exemplo de transformação simples, considerando a média das amostras de um bloco

Neste caso, as amostras transformadas $y(k_1, k_2)$ com m ou n ímpares contêm detalhes de alta resolução que, dependendo da acuidade do observador, podem ser descartados. Assim, esta transformação tão simples já é capaz de expor potenciais formas de irrelevância na imagem. Como a maioria das imagens possuem substanciais regiões suaves, espera-se que grande quantidade das amostras de detalhe, $y(k_1, k_2)$ com m ou n ímpares, tenham valores muito pequenos, senão zero. Como consequência, o índice de quantização correspondente ao valor zero também ocorrerá com alta probabilidade, devendo-se atribuir códigos simples capazes de explorar pelo menos parte desta redundância.

Uma questão chave do padrão JPEG2000 [7], por exemplo, é a sua escalabilidade. Um *bit-stream* escalonável é aquele que pode ser parcialmente descartado para se obter uma representação eficiente da imagem original ou uma versão com resolução mais baixa da imagem em taxas diferentes. Para que um sistema como esse possa ser eficiente, é necessária uma separação adequada das informações relevantes e irrelevantes de tal sorte que, quanto

maior a necessidade em se descartar amostras, formando um *bit-stream* cada vez menor, deve-se ir descartando as informações em ordem decrescente de irrelevância. Portanto, provendo uma representação escalonável da imagem de interesse.

2.5 Modelagem

Nesse trabalho, apenas serão consideradas transformações finitas e lineares, isto é, transformações que mapeiam um vetor de entrada n -dimensional \mathbf{x} em um sinal de saída m -dimensional \mathbf{y} , de acordo com:

$$\mathbf{y} = \mathbf{A}\mathbf{x}, \quad \mathbf{x} = \begin{bmatrix} x_0 \\ x_1 \\ \vdots \\ x_{n-1} \end{bmatrix}, \quad \mathbf{y} = \begin{bmatrix} y_0 \\ y_1 \\ \vdots \\ y_{m-1} \end{bmatrix}, \quad (2.20)$$

em que \mathbf{A} é uma matriz $m \times n$ com coeficientes reais ou complexos. Um exemplo de transformada complexa muito conhecido é a Transformada Discreta de Fourier (DFT - do inglês, *Discrete Fourier Transform*). De outra maneira, dada a modelagem de operação de transformada, também é possível entender essa operação como um sistema MIMO (do inglês, *Multi-Input Multi-Output*).

Restringiremos o escopo às matrizes de transformação inversíveis, tal que:

$$\mathbf{x} = \mathbf{S}\mathbf{y}$$

em que $\mathbf{S}_{n \times m}$ é a inversa generalizada da matriz \mathbf{A} , isto é, $\mathbf{S}\mathbf{A} = \mathbf{I}$, a matriz identidade de ordem $n \times n$. Logo, \mathbf{S} não é necessariamente única. No entanto, nos concentraremos em transformações não-expansivas, em que $m = n$. Nesse caso, \mathbf{S} se torna única. Observe que os coeficientes transformados podem ser expressos da seguinte maneira:

$$\mathbf{y}_q = \mathbf{a}_q \mathbf{x}, \quad q = 0, 1, \dots, m-1 \quad (2.21)$$

em que \mathbf{a}_q é a q -ésima linha da matriz $m \times m$ \mathbf{A} . Diz-se que \mathbf{a}_q é o q -ésimo vetor de análise, já que esse analisa o vetor original \mathbf{x} para compor o q -ésimo coeficiente transformado. Por extensão, chama-se \mathbf{A} de matriz de análise. A transformada inversa também pode ser expressa por:

$$\sum_{q=0}^{m-1} \mathbf{y}_q \mathbf{s}_q \quad (2.22)$$

em que \mathbf{s}_q é a q -ésima linha da matriz $m \times n$ \mathbf{S} . Diz-se que \mathbf{s}_q é o q -ésimo vetor de síntese, dado que \mathbf{x} é sintetizado por uma combinação linear de \mathbf{s}_q , tendo como pesos os coeficientes transformados. Por extensão, chama-se \mathbf{S} de matriz de síntese. Também é comum se referir a cada \mathbf{s}_q como vetor da base de síntese porque, necessariamente, o conjunto desses forma uma base do espaço linear de todos os possíveis vetores de entrada. Em resumo, os vetores de análise e de síntese são as colunas das matrizes de análise (\mathbf{A}) e de síntese (\mathbf{S}), respectivamente. Os vetores de síntese formam uma base de protótipos do sinal e a transformada decompõe sua entrada em uma combinação linear desses protótipos. Daremos então um exemplo simples e prático. A DFT, já mencionada, é definida pelas seguintes equações:

$$y_q = \frac{1}{\sqrt{m}} \sum_{p=0}^{m-1} x_p e^{-j \frac{2\pi}{m} pq}, \quad 0 \leq q < m$$

$$x_p = \frac{1}{\sqrt{m}} \sum_{q=0}^{m-1} y_q e^{j \frac{2\pi}{m} pq}, \quad 0 \leq p < m$$

Nesse caso, os vetores de análise e de síntese são iguais, os elementos $\mathbf{a}_q = \mathbf{s}_q$ são amostras do espaço unitário das senóides com frequência $f_q = \frac{q}{m}$. Em outras palavras, a DFT decompõe o vetor de entrada em uma combinação linear de exponenciais complexas, com frequências uniformemente espaçadas.

2.5.1 Transformadas separáveis

Da perspectiva de banco de filtros, existem dois modos de se chegar à construção de transformadas bidimensionais: método separável e o não separável. O jeito mais natural de se construir uma transformada bidimensional \mathbf{A} é fazendo-se uma extensão separável de uma transformada unidimensional \mathbf{A}' . Cada vetor de análise \mathbf{a}_{q_1, q_2} de uma transformada separável pode ser obtido pelo produto tensorial dos vetores unidimensionais de análise \mathbf{a}'_{q_1} e \mathbf{a}'_{q_2} . De maneira similar, também poderiam ser obtidos os vetores de síntese. A vantagem mais evidente das transformadas separáveis é que podem ser implementadas aplicando-se primeiramente a transformação unidimensional às linhas e, em seguida, às colunas ou vice-versa. Esse procedimento é ilustrado na Figura 2.9. Além disso, transformadas separáveis envolvem redução substancial na complexidade. De uma maneira simples, para se obter cada coeficiente de uma transformada não-separável seriam necessárias m^2 multiplicações e adições. Diferentemente, quando a implementação for separável, apenas seriam necessárias $2m$ multiplicações e adições.

Pode-se então representar matricialmente uma transformada bidimensional separável, de forma análoga à equação 2.20:

$$\mathbf{y} = \mathbf{A}'\mathbf{x}\mathbf{A}'^T \quad (2.23)$$

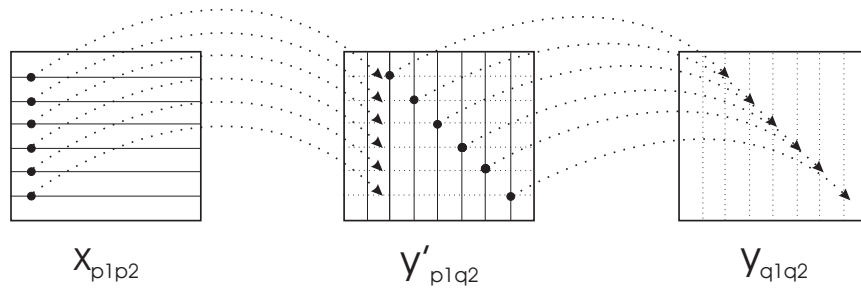


Figura 2.9: Implementação separável de uma transformação bidimensional, aplicada unidimensionalmente às linhas e, em seguida, às colunas

No entanto, deve-se notar que transformadas separáveis têm a tendência a priorizar o aproveitamento de direções horizontais, verticais e diagonais. No caso da decomposição bidimensional separável por *wavelets*, um arranjo tradicional das sub-bandas é, inclusive, o que evidencia a sub-banda de mais baixa frequência, que é considerada uma aproximação (A) da imagem original, ao lado das sub-bandas que representam os detalhes verticais (V), horizontais (H) e diagonais (D), como ilustrado na Figura 2.10. Nessa figura, também é feita a correspondência do tipo de detalhe evidenciado para cada sequência de filtragens passa-baixa, passa-alta realizada. Por exemplo, considerando-se que essa transformação tenha sido realizada primeiramente ao longo das linhas e depois ao longo das colunas de uma imagem, a sequência de filtragem LH (L, do inglês, *Low-Pass*; e H, do inglês *High-pass*), seria responsável por privilegiar o aparecimento de detalhes verticais, visto que as alta frequências das colunas foram preservadas. De maneira análoga, pode-se relacionar as demais sub-bandas.

Uma solução alternativa é a de se utilizar bancos de filtros não separáveis, cujo exemplo bastante conhecido é o banco de filtros *quincunx* [8][9]. Nesse contexto multidimensional, o termo não separável pode se referir tanto aos filtros, quanto ao tipo de amostragem dos dados.

Cada um desses métodos apresenta vantagens e também desvantagens. Enquanto os filtros separáveis oferecem baixa complexidade de processamento, os não separáveis apresentam maiores graus de liberdade e, portanto, projetos melhores. Desde já torna-se importante destacar que a transformada direcional implementada nesse trabalho possui natureza não separável. Esse assunto será retomado no Capítulo 4.

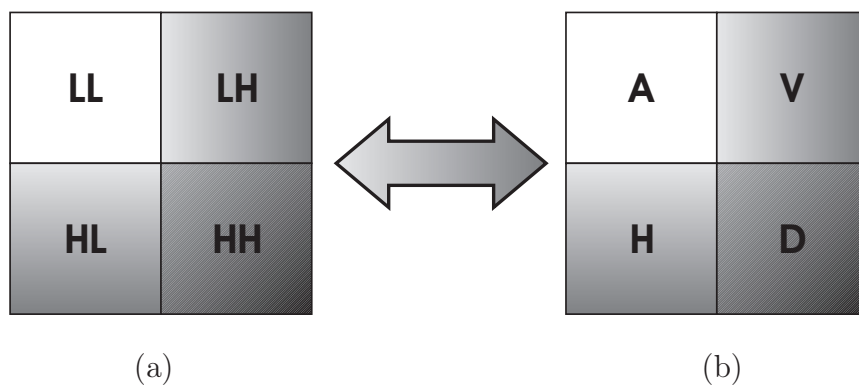


Figura 2.10: (a) das sub-bandas de uma transformada *wavelet* bidimensional: LL - simbolizando duas filtragens passa-baixa (do inglês, *Low-pass*); LH - simbolizando uma filtragem passa-baixa (do inglês, *Low-pass*), seguida de uma filtragem passa-alta (do inglês, *High-pass*); HL - simbolizando uma filtragem passa-alta (do inglês, *High-pass*), seguida de uma filtragem passa-baixa (do inglês, *Low-pass*); HH - simbolizando duas filtragens passa-alta (do inglês, *High-pass*). (b) Organização evidenciando os detalhes destacados em cada sub-banda: A - aproximação; V - detalhe vertical; H - detalhe horizontal; D - detalhe diagonal.

Capítulo 3

Bancos de Filtros

3.1 Banco de filtros e reconstrução perfeita

Um banco de filtros nada mais é do que um arranjo de filtros, e estudar esses sistemas como um todo significa se importar, fundamentalmente, com duas questões: reconstrução sem *aliasing*, [10],[11],[12] e reconstrução perfeita. Esse sistema é utilizado para separar um sinal de entrada em vários componentes, cada um contendo uma faixa de frequência do sinal original. É igualmente desejado que o projeto desse banco permita que o sinal original possa ser reconstruído a partir de seus componentes passa-faixa. O primeiro processo é chamado análise, enquanto o segundo é chamado síntese [10].

Do ponto de vista sistêmico, um banco de filtros digital com M canais pode ser entendido como um arranjo de M filtros digitais com uma entrada em comum, o que corresponde ao processo de análise, ou com uma saída em comum, o que corresponde ao processo de síntese. Ambos os casos são ilustrados na Figura 3.1. Assim, o número de bandas (canais ou faixas) de um banco de filtros nada mais é do que a quantidade de filtros que compõe cada banco. O sistema da Figura 3.1 (a) é chamado banco de filtros de análise, e os filtros $H_k(z)$ são os filtros de análise. Esse sistema separa o sinal de entrada $x(n)$ em M sinais passa-faixa $x_k(n)$. Já o sistema da Figura 3.1 (b) é chamado banco de filtros de síntese, e os filtros $F_k(z)$ são os filtros de síntese. Esse outro sistema então combina os M sinais passa-faixa em um sinal $\hat{x}(n)$, que é, em princípio, uma aproximação do sinal de entrada $x(n)$.

O intervalo que separa o bancos de filtro de análise do de síntese indica a possibilidade de os sinais passa-faixa serem codificados ou transmitidos até que sejam sintetizados. Nesse ponto, ressalta-se a existência de fonte de erro na reconstrução, que é afeta à codificação e à quantização dos sinais subamostrados. Os sinais de saída do banco de análise podem ser codificados por várias técnicas de compressão e em seguida poderão ser transmitidos. Fundamentalmente, no processo de compressão os sinais serão quantizados, inserindo

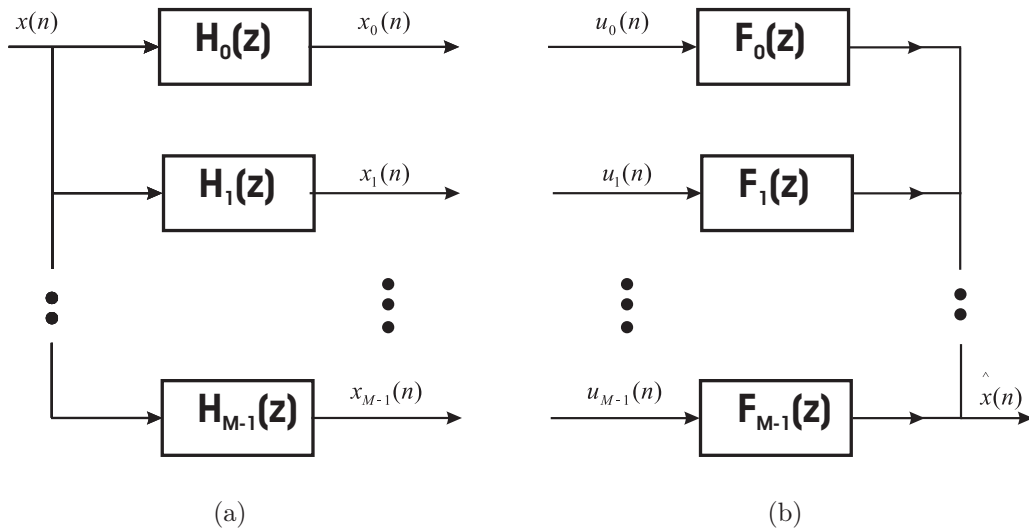


Figura 3.1: Banco de Filtros Digitais: (a) banco de análise; (b) banco de síntese.

uma distorção não linear no sistema. Dessa maneira, no receptor serão reconstruídas aproximações desses sinais. O erro de decodificação, obtido pela diferença entre os sinais transmitidos e os reconstruídos, é chamado *erro de quantização de subbandas*. Como o assunto compressão com perdas já foi abordado no Capítulo 2, toda discussão sobre reconstrução perfeita abrangerá apenas condições sobre os filtros componentes dos bancos, isto é, assumiremos que o intervalo não existe e que, portanto, não existem perdas associadas à compressão na conexão entre os bancos de análise e de síntese.

Uma maneira simples de se definir um banco de filtros consiste em discriminar seus componentes, isto é, um banco de filtros é um conjunto de filtros, unidos por operadores de amostragem e, às vezes, também por operadores que geram atrasos. A Figura 3.2 mostra a configuração de um banco de filtros com M bandas.

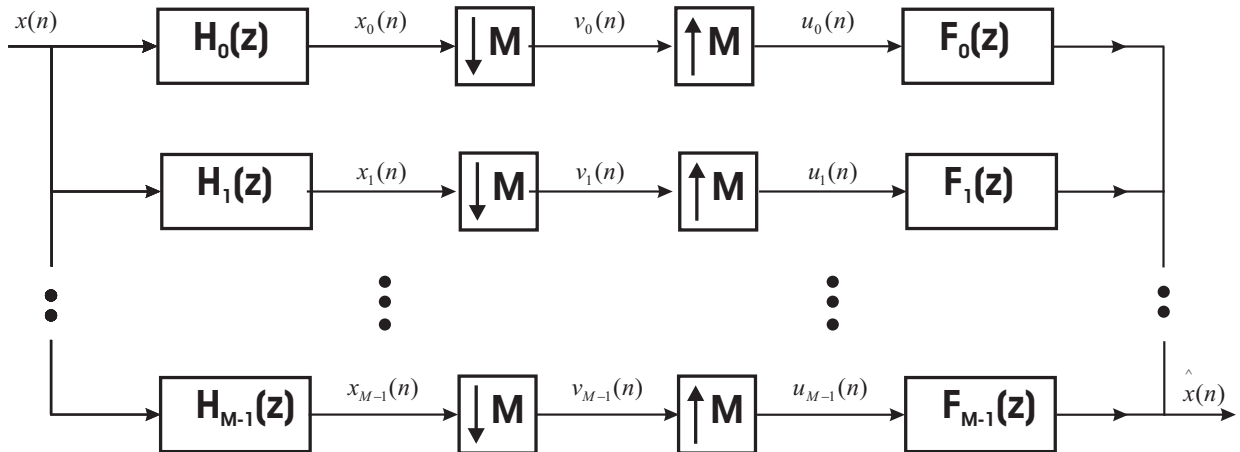


Figura 3.2: Banco de Filtros com M faixas criticamente decimado.

Os operadores de amostragem são responsáveis por alterar a taxa de amostragem por um fator inteiro e, portanto, o número de amostras do sinal de entrada. Logo, esses operadores

podem subamostrar o sinal, descartando parte de suas amostras e, portanto, reduzindo a taxa de amostragem, quando serão chamados decimadores ou dizimadores (processo conhecido em inglês como *downsampling*); ou poderão sobreamostrar o sinal, inserindo novas amostras e, portanto, aumentando a taxa de amostragem, quando serão chamados expansores ou sobreamostradores (processo conhecido em inglês como *upsampling*). A Figura 3.3 mostra os símbolos para essas operações, bem como a relação entre os sinais de entrada e de saída. Desde já, pode ser evidenciada a importância desses operadores. Observando-se o arranjo da Figura 3.1 nota-se que cada um dos sinais passa-faixa x_k terá, no mínimo, o mesmo número de amostras do sinal original, já que a operação de filtragem é uma operação de convolução no domínio temporal. Logo, caso não existissem operadores para descartar amostras, na saída do banco de análise teriam-se M vezes a quantidade de amostras do sinal original. Como esses sinais passa-faixa, em regra, serão armazenados ou transmitidos, essa expansão é notadamente indesejável, visto que mídias com maior capacidade e que maiores faixas de transmissão implicam em custos mais altos. Se assumirmos que os filtros são ideais e que cada sinal passa-faixa tem, portanto, largura de banda de π/M rads, cada um desses sinais pode ser subamostrado de $M : 1$ sem perda de informação. Esse é o princípio dos bancos de filtros criticamente decimados [10],[13], [14].

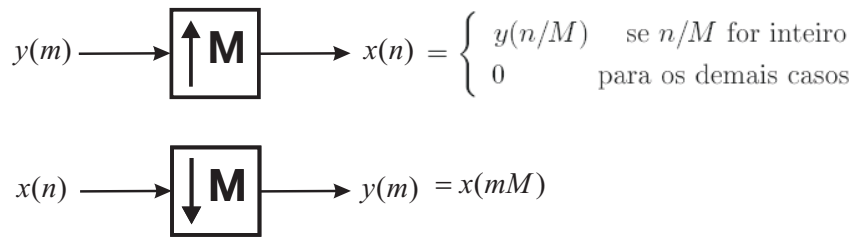
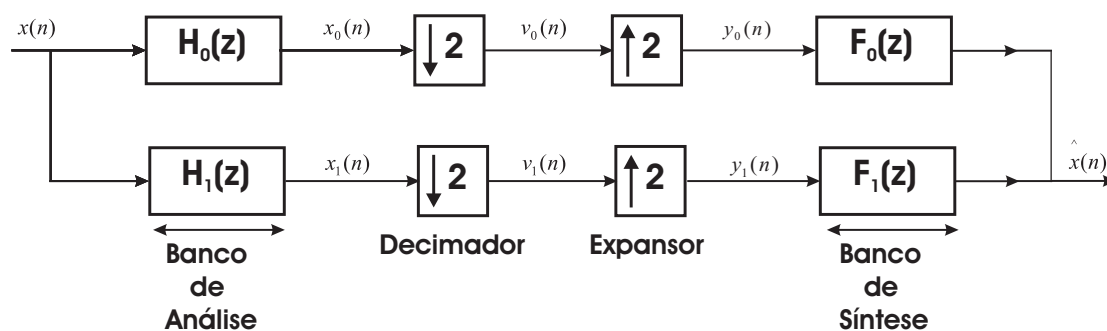


Figura 3.3: Operações básicas para a conversão da taxa de amostragem. O sobreamostrador ou expensor (a) insere $M - 1$ amostras nulas entre cada par de amostras originais, aumentando a taxa de amostragem por um fator M . O decimador (b) retém uma a cada M amostras do sinal original, reduzindo a taxa de amostragem por um fator M .

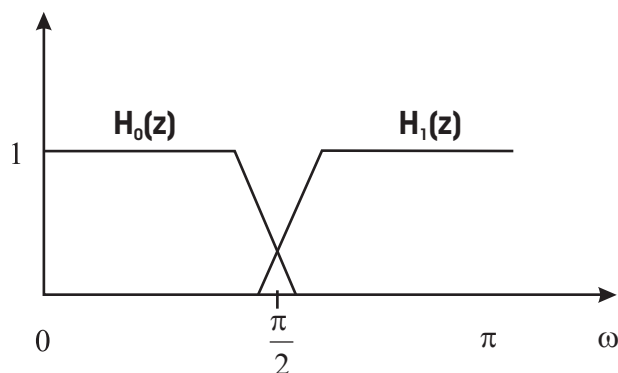
Inicialmente, pode-se então fazer o número de faixas do banco de filtros da Figura 3.1 igual a 2, $M = 2$, simplificando-se a estrutura geral e obtendo-se a Figura 3.4.

Nessa estrutura, popularmente chamada banco de filtros espelhados em quadratura (QMF, do inglês *Quadrature Mirror Filters*)[15], o sinal de entrada $x(n)$ é filtrado por dois filtros H_0 e H_1 , que são tipicamente filtros passa-baixa e passa-alta, como mostrado na parte (b) da Figura 3.4. Além disso, o filtro de análise passa-alta é projetado de forma a reverter os sinais de resposta ao impulso:

$$H_1(z) = H_0(-z). \quad (3.1)$$



(a)



(b)

Figura 3.4: (a) Banco de Filtros Espelhados em Quadratura;(b) repostas típicas de magnitude.

Após o sinal digital de entrada ter sido decomposto em duas faixas de frequência, os sinais em cada faixa $x_k(n)$, para $k = 0, 1$, serão então subamostrados por um fator de 2 para produzir os sinais $v_k(n)$.

Cada sinal decimado $v_k(n)$ será então codificado de maneira que possam ser aproveitadas características especiais desses sinais, como, por exemplo, nível de energia ou, até mesmo, a sensibilidade do receptor. No receptor final, os sinais recebidos serão decodificados, produzindo aproximações dos sinais $v_0(n)$ e $v_1(n)$, que serão expandidos ou superamostrados por um fator de 2. Os sinais resultantes $y_0(n)$ e $y_1(n)$ serão filtrados por $F_0(z)$ e $F_1(z)$ para dar origem ao sinal de saída $\hat{x}(n)$.

Os filtros H_0 e H_1 indicados no início da estrutura são chamados filtros de análise, e o par $[H_0 H_1]$ é conhecido como banco de filtros de análise. Esse par, seguido dos subamostradores $\downarrow (2)$ forma o chamado *banco de filtros de análise decimado*. De forma análoga, os filtros F_0 e F_1 são os filtros de síntese ou de reconstrução, e o par $[F_0 F_1]$ é o banco de filtros de síntese. É importante notar que o sinal reconstruído $\hat{x}(n)$ difere do sinal original $x(n)$ por três principais motivos: *aliasing*, distorção de amplitude e distorção de fase. No entanto, essa seção tem por finalidade esclarecer que ainda assim é possível se obter reconstrução perfeita, desde que o banco de reconstrução seja projetado para anular os erros inseridos

pelo banco de análise. Note-se que para o caso de um banco de filtros de M faixas as condições de projeto para reconstrução perfeita são bem mais complicadas. Uma técnica para o projeto de sistemas de M faixas com reconstrução perfeita foi desenvolvida [16], [17], baseada na propriedade paraunitária de matrizes polifásicas.

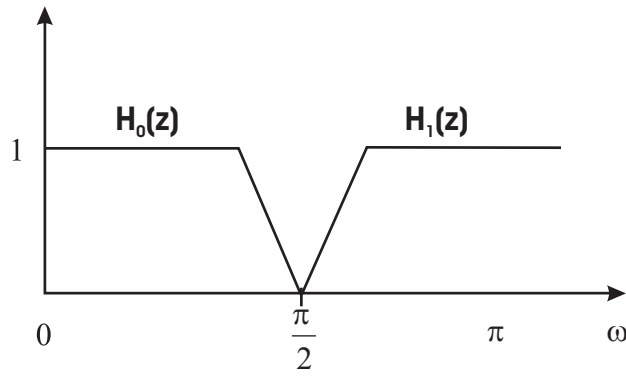
3.2 Análise dos erros gerados em um banco QMF

Nessa seção, serão melhor analisadas as três principais fontes de erros gerados em bancos de filtros QMF: *aliasing* e *imaging*, distorções de amplitude e de fase.

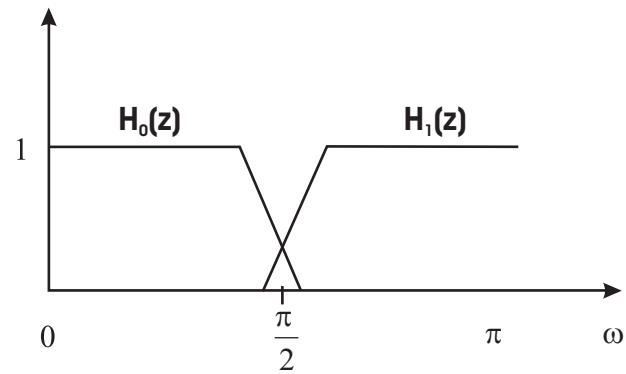
3.2.1 Erros de *aliasing* e de *imaging*

A Figura 3.4 mostra um esboço da resposta em frequência dos filtros de análise. Como apenas filtros factíveis serão considerados, logo, não-ideais, vê-se que a resposta dos filtros se sobrepõem pois existe uma faixa de transição e um pequeno ganho na faixa de rejeição. Assim, os sinais $x_k(n)$ não serão limitados em cada uma das bandas e a operação de decimação gerará *aliasing*. Para que esse efeito seja melhor entendido, considere a Figura 3.5, que ilustra duas situações interessantes.

Na Figura 3.5 (a), as respostas $|H_1(e^{j\omega})|$ e $|H_0(e^{j\omega})|$ não se sobrepõem. Assumindo que a atenuação da banda rejeição de cada um dos filtros seja suficientemente grande, o efeito de *aliasing* não será tão sério. No entanto, na Figura 3.5 (b) as respostas dos dois filtros se sobrepõem e cada um dos filtros tem energia suficiente na faixa de frequência que excede à ideal. Nesse caso, a decimação dos sinais causará *aliasing* independentemente de quão atenuadas sejam as faixas de rejeição dos filtros. Em princípio, a escolha sem superposição da Figura 3.5(a) evitaria o *aliasing*. No entanto, respostas não sobrepostas implicariam em uma forte atenuação do sinal de entrada em torno de $\omega = \frac{\pi}{2}$. Pode-se então pensar que pré-amplificar os componentes do sinal de entrada nessa faixa de frequência compensaria a excessiva atenuação. Porém, isso também amplificaria ruídos (como o ruído de codificação ou o ruído de canal), o que seria indesejável. Uma outra solução seria tornar cada um dos filtros com a mínima faixa de transição possível, porém isso tornaria o filtros excessivamente grandes. A resposta sobreposta apresenta-se, então, como a solução mais prática, mesmo que cause *aliasing*, visto que esse efeito ainda pode ser anulado por um projeto adequado dos filtros de síntese, como será mostrado.



(a)



(b)

Figura 3.5: Duas possíveis repostas em magnitude para os filtros de análise: (a) sem sobreposição, e (b) com sobreposição.

Expressão matemática do sinal reconstruído

Usando resultados conhecidos da área de análise de sistemas multitaxa, se chegará a uma expressão para descrever o sinal reconstruído $\hat{x}(n)$. A partir da Figura 3.4 (a), tem-se:

$$X_k(z) = H_k(z)X(z), \quad k = 0, 1 \quad (3.2)$$

A transformada z dos sinais decimados $v_k(n)$ pode ser escrita em função de $X_k(z)$:

$$V_k(z) = \frac{1}{2}[X_k(z^{\frac{1}{2}}) + X_k(-z^{\frac{1}{2}})], \quad k = 0, 1 \quad (3.3)$$

O segundo termo da equação acima representa o efeito de *aliasing*, que se deseja eliminar. A transformada z do $Y_k(z)$ é $V_k(z^2)$, então:

$$Y_k(z) = V_k(z^2) = \frac{1}{2}[X_k(z) + X_k(-z)], \quad k = 0, 1 \quad (3.4)$$

O sinal reconstruído $\hat{X}(z)$ é:

$$\hat{X}(z) = F_0(z)Y_0(z) + F_1(z)Y_1(z). \quad (3.5)$$

Substituindo (3.4) em (3.5) e rearranjando os termos, vem:

$$\hat{X}(z) = \frac{1}{2}[H_0(z)F_0(z) + H_1(z)F_1(z)]X(z) + \frac{1}{2}[H_0(-z)F_0(z) + H_1(-z)F_1(z)]X(-z). \quad (3.6)$$

Em notação matricial:

$$2\hat{X}(z) = \begin{bmatrix} X(z) & X(-z) \end{bmatrix} \underbrace{\begin{bmatrix} H_0(z) & H_1(z) \\ H_0(-z) & H_1(-z) \end{bmatrix}}_{\mathbf{H}(z)} \begin{bmatrix} F_0(z) \\ F_1(z) \end{bmatrix}. \quad (3.7)$$

A matriz $\mathbf{H}(z)$ é chamada matriz de componentes de *aliasing*. O termo que contém $X(-z)$ é originado pelos decimadores. Sobre o círculo unitário, $X(-z) = X(e^{j(\omega-\pi)})$, que representa um deslocamento do espectro de π para a direita. Esse termo representa tanto o efeito de *aliasing*, provocado pelos decimadores, quanto o efeito de *imaging*, provocado pelos sobreamostradores.

Cancelamento de *aliasing*

A partir da equação 3.6, pode-se perceber que é possível cancelar o efeito de *aliasing* projetando-se adequadamente os filtros de síntese $F_0(z)$ e $F_1(z)$ para que $H_0(-z)F_0(z) + H_1(-z)F_1(z)$ seja zero. Então, pode-se ver que a seguinte escolha é adequada ao cancelamento:

$$F_0(z) = H_1(-z), \quad F_1(z) = -H_0(-z) \quad (3.8)$$

Logo, dados H_0 e H_1 é completamente possível anular-se o efeito de *aliasing* escolhendo-se os filtros de síntese como em (3.8). Assim, ainda que os filtros de análise tenham grandes faixas de transição e baixa atenuação nas faixas de rejeição, o que provoca grande efeito de *aliasing*, esses erros serão cancelados pela escolha adequada dos filtros de síntese.

A idéia básica do projeto de banco de filtros QMF é a de permitir a geração do efeito de *aliasing* no banco de filtros de análise ao invés de tentar evitá-lo. Segue-se então o projeto do banco de filtros de síntese para que o componente de *aliasing* do canal passa-baixa de análise seja cancelado pelo canal passa-alta de síntese e vice-versa.

3.2.2 Distorção de amplitude e de fase

Inicialmente será feita uma breve discussão sobre os conceitos de distorção de amplitude e de fase baseada na teoria de sistemas lineares.

A distorção de amplitude ocorre quando a resposta em módulo do sistema não é constante dentro das faixas de frequência de interesse. Nesse caso, os diferentes componentes em frequência do sinal de entrada receberão diferentes valores de ganho ou de atenuação. Já a distorção de fase é caracterizada por provocar retardos diferentes aos componentes de frequência, dado que a resposta em fase do sistema não é linear. Como consequência, o sinal de saída terá uma forma de onda diferente da do sinal de entrada.

Da teoria de sistemas multitaxa, sabe-se que os operadores de subamostragem e de sobreamostragem são lineares, mas variantes ao deslocamento [10]. Desta maneira, em princípio, um banco de filtros é também um sistema linear variante ao deslocamento. Além disso, já foi visto que esses operadores também são responsáveis por gerar efeitos de *aliasing* e de *imaging*. No entanto, se considerarmos que o banco de filtros QMF foi projetado de maneira a se obter o cancelamento do efeito de *aliasing*, o sistema globalmente deixará de ser variante ao deslocamento. Dessa maneira, o sistema passará a ser representado por uma função de transferência $T(z)$, tal que:

$$\hat{X}(z) = T(z)X(z). \quad (3.9)$$

Assim, ainda que o sistema esteja livre do efeito de *aliasing*, o sinal $\hat{x}(n)$ sofrerá de distorção linear e invariante a deslocamento $T(z)$. No caso do banco de filtros QMF, tem-se:

$$T(z) = \frac{1}{2}[H_0(z)F_0(z) + H_1(z)F_1(z)]. \quad (3.10)$$

Substituindo-se 3.8 em 3.11, tem-se:

$$T(z) = \frac{1}{2}[H_0(z)H_1(-z) - H_1(z)H_0(-z)]. \quad (3.11)$$

Avaliando-se $T(z)$ sobre o círculo unitário, $T(e^{j\omega}) = |T(e^{j\omega})|e^{j\phi(\omega)}$, tem-se:

$$\hat{X}(e^{j\omega}) = |T(e^{j\omega})|e^{j\phi(\omega)}X(e^{j\omega}). \quad (3.12)$$

Os conceitos apresentados no início da seção podem então ser aplicados ao sistema com a função de transferência acima. Assim, a menos que $T(z)$ seja um filtro passa-tudo, isto é, $|T(e^{j\omega})| = d \neq 0$ para todo ω , diz-se que $\hat{X}(e^{j\omega})$ apresentará distorção de amplitude. Analogamente, a não ser que $T(z)$ tenha fase linear, isto é, $\phi(\omega) = a + b\omega$ para a e b constantes, $\hat{X}(e^{j\omega})$ também apresentará distorção de fase.

3.2.3 Um sistema QMF simples com cancelamento de *alias*

Em um dos mais antigos bancos de filtros QMF os filtros de análise se relacionam da seguinte maneira:

$$H_1(z) = H_0(-z). \quad (3.13)$$

Com a restrição de projeto 3.8 para se obter o cancelamento de *alias*, e com a escolha 3.13, os outros filtros serão assim determinados:

$$F_0(z) = H_0(z), \quad F_1(z) = -H_1(z) = -H_0(-z). \quad (3.14)$$

Nota-se que, nesse caso, os quatro filtros serão determinados pela escolha de um único filtro $H_0(z)$. Com os filtros escolhidos da maneira descrita, obtém-se a seguinte função de distorção:

$$T(z) = \frac{1}{2}(H_0^2(z) - H_1^2(z)) = \frac{1}{2}(H_0^2(z) - H_0^2(-z)). \quad (3.15)$$

3.2.4 Representação Polifásica

Normalmente é conveniente, conceitual e computacionalmente, representar os bancos de análise e de síntese em função de suas componentes polifásicas. Para que se possa entender a idéia básica dessa representação, considere um filtro $H(z) = \sum_{n=-\infty}^{\infty} h(n)z^{-n}$. Separando-se os coeficientes pares dos ímpares, vem:

$$H(z) = \sum_{n=-\infty}^{\infty} h(2n)z^{-2n} + z^{-1} \sum_{n=-\infty}^{\infty} h(2n+1)z^{-2n}. \quad (3.16)$$

Pode-se definir:

$$E_0(z) = \sum_{n=-\infty}^{\infty} h(2n)z^{-n}, \quad E_1(z) = \sum_{n=-\infty}^{\infty} h(2n+1)z^{-n}. \quad (3.17)$$

Pode-se, então, reescrever o filtro em função de suas componentes polifásicas E_0 e E_1 :

$$H(z) = E_0(z^2) + z^{-1}E_1(z^2). \quad (3.18)$$

Considere, como exemplo, um banco com os seguintes filtros de análise: $H_0(z) = h_0 + h_1z^{-1} + h_2z^{-2} + h_3z^{-3}$ e $H_1(z) = -h_3z^2 + h_2z^1 - h_1 + h_0z^{-1}$. Cada um desses filtros poderia

então ser decomposto em suas polifásicas. Primeiramente, pode-se decompor H_0 :

$$E_{00}(z^2) = h_0 + h_2z^{-2}, \quad E_{01}(z^2) = h_1 + h_3z^{-2}. \quad (3.19)$$

Em seguida, H_1 será decomposto:

$$E_{10}(z^2) = -h_3z^2 - h_1, \quad E_{11}(z^2) = h_2z^2 + h_0. \quad (3.20)$$

Por fim, os componentes polifásicos dos filtros de análise comporão a matriz que mais tarde será chamada de matriz polifásica de análise:

$$\mathbf{E}(\mathbf{z}) = \begin{bmatrix} E_{00}(z) & E_{01}(z) \\ E_{10}(z) & E_{11}(z) \end{bmatrix} = \begin{bmatrix} h_0 + h_2z^{-1} & h_1 + h_3z^{-1} \\ -h_3z^1 - h_1 & h_2z^1 + h_0 \end{bmatrix}. \quad (3.21)$$

Observa-se que, ao serem considerados os dois filtros do banco de análise, inseriu-se no subscrito do identificador da componente polifásica o número correspondente ao filtro decomposto. Dessa maneira, o elemento E_{00} indica a polifásica E_0 do filtro H_0 , enquanto E_{10} indica a mesma polifásica E_0 , porém do filtro H_1 .

Desde já, pode-se então estender o conceito de decomposição em polifásicas a um banco de filtros com M faixas, que será amplamente discutido na seção 3.3:

$$\begin{aligned} H(z) &= \sum_{n=-\infty}^{\infty} h(nM)z^{-nM} \\ &+ z^{-1} \sum_{n=-\infty}^{\infty} h(nM + 1)z^{-nM} \\ &\vdots \\ &+ z^{-(M-1)} \sum_{n=-\infty}^{\infty} h(nM + M - 1)z^{-nM}. \end{aligned} \quad (3.22)$$

De maneira mais compacta:

$$H(z) = \sum_{l=0}^{M-1} z^{-l} E_l(z^M), \quad (3.23)$$

em que:

$$E_l(z) = \sum_{n=-\infty}^{\infty} e_l(n)z^{-n}, \quad (3.24)$$

com:

$$e_l(n) \triangleq h(Mn + l), \quad 0 \leq l \leq (M - 1). \quad (3.25)$$

Após a breve discussão sobre a representação polifásica de um filtro, aplicaremos então essa decomposição aos filtros de análise de um banco QMF. Assim, começaremos pelo filtro passa-baixa de análise:

$$H_0(z) = E_0(z^2) + z^{-1}E_1(z^2). \quad (3.26)$$

Dado que $H_1(z) = H_0(-z)$, tem-se $H_1(z) = E_0(z^2) - z^{-1}E_1(z^2)$. Logo:

$$\begin{bmatrix} H_0(z) \\ H_1(z) \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} E_0(z^2) \\ z^{-1}E_1(z^2) \end{bmatrix}. \quad (3.27)$$

3.2.5 Limitações impostas pela restrição $H_1(z) = -H_0(z)$, em um banco QMF

Relacionando-se os filtros de análise de forma que $H_1(z) = -H_0(z)$, e escolhendo-se os filtros de síntese para que haja o cancelamento de *aliasing*, conforme 3.8, a função de distorção tem a forma 3.15. Pode-se então reescrever essa função em termos das componentes polifásicas de cada filtro:

$$T(z) = 2z^{-1}E_0(z^2)E_1(z^2). \quad (3.28)$$

Desta maneira, se o filtro H_0 for FIR, então $E_0(z)$ e $E_1(z)$ também serão e, conseqüentemente, $T(z)$ também será. Logo, nota-se que somente será possível eliminar a distorção de amplitude se cada uma das funções FIR $E_0(z)$ e $E_1(z)$ forem atrasos puros, isto é, $E_0(z) = c_0z^{-n_0}$ e $E_1(z) = c_1z^{-n_1}$. Isso significa que os filtros serão expressos por:

$$H_0(z) = c_0z^{-2n_0} + c_1z^{-2n_1+1}, \quad H_1(z) = c_0z^{-2n_0} - c_1z^{-2n_1+1}. \quad (3.29)$$

Em resumo, no caso QMF, em que os filtros de análise se relacionam como $H_1(z) = H_0(-z)$, e quando $H_0(z)$ é um filtro FIR, a distorção de amplitude somente será eliminada se os filtros forem simples atrasadores. Como conseqüência, vê-se que esses filtros não poderão região de corte abrupta, nem grandes atenuações na faixa de rejeição, condições características de filtros ideais. No entanto, como já discutido, ainda assim é possível se obter reconstrução perfeita com a restrição $H_1(z) = H_0(-z)$.

3.3 Bancos de filtros com M faixas

Nas seções anteriores, ao se estudar o banco QMF de duas faixas, foi considerado o caso específico em que os filtros se relacionam como $H_1(z) = -H_0(z)$. Torna-se importante analisar o caso mais geral, em que, *a priori*, nenhuma restrição é imposta, para que possam ser feitas considerações a respeito de condições para cancelamento de *aliasing* e para reconstrução perfeita. Para tal estudo, será considerado o banco de filtros com M faixas criticamente decimado, ilustrado na Figura 3.2, repetida na Figura 3.6 por conveniência.

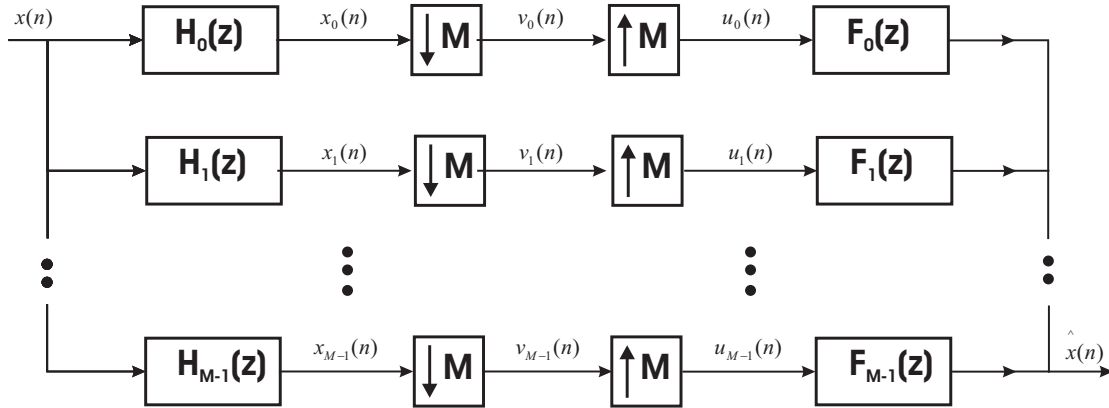


Figura 3.6: Banco de Filtros com M faixas criticamente decimado

Na Figura 3.6, o sinal $x(n)$ é separado em M sinais passa-faixa pelos M *filtros de análise* $H_k(z)$. Cada um dos sinais $x_k(n)$ é então decimado para se obter os sinais $v_k(n)$. Os sinais $v_k(n)$ finalmente serão passados pelos expansores, constituindo os sinais $u_k(n)$, e recombinaos pelos filtros $F_k(z)$ para originar o sinal reconstruído $\hat{x}(n)$.

O objetivo dessa seção é generalizar as importantes propriedades anteriormente discutidas para um banco QMF com $M = 2$, sobretudo às relacionadas ao cancelamento de *aliasing* e à reconstrução perfeita. Para tal, será conveniente definir os seguintes vetores:

$$\mathbf{h}(z) = \underbrace{\begin{bmatrix} H_0(z) \\ H_1(z) \\ \vdots \\ H_{M-1}(z) \end{bmatrix}}_{\text{Banco de Análise}}, \quad \mathbf{f}(z) = \underbrace{\begin{bmatrix} F_0(z) \\ F_1(z) \\ \vdots \\ F_{M-1}(z) \end{bmatrix}}_{\text{Banco de Síntese transposto}}, \quad \mathbf{e}(z) = \underbrace{\begin{bmatrix} 1 \\ z^{-1} \\ \vdots \\ z^{M-1} \end{bmatrix}}_{\text{Cadeia de atrasos}}, \quad (3.30)$$

Note que o banco de análise é um sistema com uma única entrada e M saídas, representado pela matriz de transferência $\mathbf{h}(z)$; já o banco de síntese é um sistema com M entradas e uma única saída, representado pela matriz de transferência $\mathbf{f}^T(z)$. O vetor denominado cadeia de atrasos será utilizado nas representações polifásicas, assunto discutido no tópico correspondente da seção anterior.

3.3.1 Expressão matemática do sinal reconstruído - M faixas

Primeiramente, o sinal reconstruído $\hat{x}(n)$ será escrito em termos de $x(n)$, ignorando-se a presença de erros de codificação e de quantização. Dessa maneira, cada sinal passa-faixa, resultante da filtragem do sinal de entrada pelo respectivo filtro de análise, é obtido por:

$$X_k(z) = H_k(z)X(z). \quad (3.31)$$

Então, utilizando-se propriedade conhecida em sistemas multitaxa, os sinais decimados terão a seguinte transformada z :

$$V_k(z) = \frac{1}{M} \sum_{l=0}^{M-1} H_k(z^{1/M}W^l)X(z^{1/M}W^l), \quad (3.32)$$

em que $W = W_M = e^{-j2\pi/M}$. A saída dos expansores será dada por:

$$U_k(z) = V_k(z^M) = \frac{1}{M} \sum_{l=0}^{M-1} H_k(zW^l)X(zW^l). \quad (3.33)$$

Assim, o sinal reconstruído será:

$$\hat{X}(z) = \sum_{k=0}^{M-1} F_k(z)U_k(z) = \frac{1}{M} \sum_{l=0}^{M-1} X(zW^l) \sum_{k=0}^{M-1} H_k(zW^l)F_k(z). \quad (3.34)$$

Essa equação pode ser reescrita de maneira mais conveniente:

$$\hat{X}(z) = \sum_{l=0}^{M-1} A_l(z), \quad (3.35)$$

em que

$$A_l(z) = \frac{1}{M} \sum_{k=0}^{M-1} H_k(zW^l)F_k(z), \quad 0 \leq l \leq M-1. \quad (3.36)$$

Para $z = e^{j\omega}$, $X(zW^l)$ pode ser escrito como:

$$X(e^{j\omega}W^l) = X(e^{j(\omega - \frac{2\pi l}{M})}). \quad (3.37)$$

A partir da equação acima, para $l \neq 0$, vê-se que $X(e^{j\omega}W^l)$ representa uma versão deslocada de $X(e^{j\omega})$. Portanto, considerando a informação em (3.34), pode-se dizer que o espectro do sinal reconstruído $\hat{X}(e^{j\omega})$ é uma combinação linear de $X(e^{j\omega})$ e suas $M-1$ versões uniformemente deslocadas.

3.3.2 Representação Polifásica - M Faixas

Na seção 3.2.4 foi introduzido o conceito de representação polifásica, muito útil tanto para estudos teóricos quanto para implementação prática. Nessa mesma seção, foi visto que qualquer função de transferência $H_k(z)$ pode ser expressa em função de suas componentes polifásicas como:

$$H_k(z) = \sum_{l=0}^{M-1} E_{kl}(z^M) \quad (3.38)$$

Em notação matricial:

$$\begin{bmatrix} H_0(z) \\ \vdots \\ H_{M-1}(z) \end{bmatrix} = \begin{bmatrix} E_{00}(z^M) & E_{01}(z^M) & \dots & E_{0,M-1}(z^M) \\ \vdots & \vdots & \ddots & \vdots \\ E_{M-1,0}(z^M) & E_{M-1,1}(z^M) & \dots & E_{M-1,M-1}(z^M) \end{bmatrix} \begin{bmatrix} 1 \\ z^{-1} \\ \vdots \\ z^{-(M-1)} \end{bmatrix}. \quad (3.39)$$

Isto é:

$$\mathbf{h}(z) = \mathbf{E}(z^M)\mathbf{e}(z), \quad (3.40)$$

em que

$$\mathbf{E}(z) = \begin{bmatrix} E_{00}(z) & E_{01}(z) & \dots & E_{0,M-1}(z) \\ E_{10}(z) & E_{11}(z) & \dots & E_{1,M-1}(z) \\ \vdots & \vdots & \ddots & \vdots \\ E_{M-1,0}(z) & E_{M-1,1}(z) & \dots & E_{M-1,M-1}(z) \end{bmatrix}, \quad (3.41)$$

e $\mathbf{h}(z)$ e $\mathbf{e}(z)$ foram definidos em 3.30. Note que a matriz montada na seção 3.2.4 pode ser obtida apenas fazendo-se $M = 2$ em 3.41. A Figura 3.7 ilustra a equivalência na representação do banco de filtros de análise quando esses são substituídos pelas suas polifásicas. A matriz $\mathbf{E}(z)$ é a matriz polifásica de ordem $M \times M$ do banco de análise.

De maneira análoga, os filtros de síntese também podem ser expressos em suas componentes polifásicas:

$$F_k(z) = \sum_{l=0}^{M-1} z^{-(M-1-l)} R_{lk}(z^M). \quad (3.42)$$

Em notação matricial:

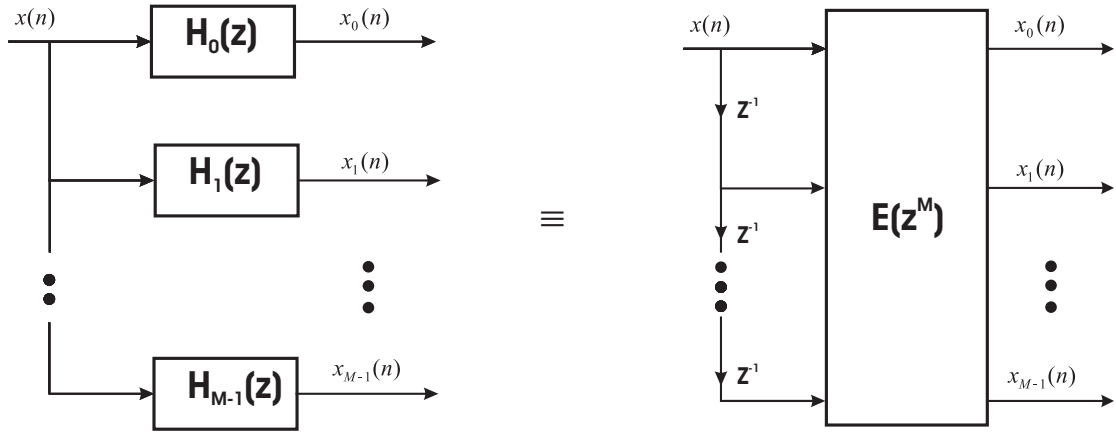


Figura 3.7: Representação polifásica de um banco de análise com M faixas

$$\begin{bmatrix} F_0(z) & \dots & F_{M-1}(z) \end{bmatrix} = \begin{bmatrix} z^{-(M-1)} & z^{-(M-2)} & \dots & 1 \end{bmatrix} \begin{bmatrix} R_{00}(z^M) & \dots & R_{0,M-1}(z^M) \\ R_{10}(z^M) & \dots & R_{1,M-1}(z^M) \\ \vdots & \ddots & \vdots \\ R_{M-1,0}(z^M) & \dots & R_{M-1,M-1}(z^M) \end{bmatrix}. \quad (3.43)$$

Em termos de $\mathbf{e}(z)$ e do vetor $\mathbf{f}^T(z)$ que representa o banco de síntese, vem:

$$\mathbf{f}^T(z) = z^{-(M-1)} \mathbf{e}^T(z) \mathbf{R}(z^M), \quad (3.44)$$

em que:

$$\mathbf{R}(z) = \begin{bmatrix} R_{00}(z) & \dots & R_{0,M-1}(z) \\ R_{10}(z) & \dots & R_{1,M-1}(z) \\ \vdots & \ddots & \vdots \\ R_{M-1,0}(z) & \dots & R_{M-1,M-1}(z) \end{bmatrix}. \quad (3.45)$$

A matriz $\mathbf{R}(z)$ é a matriz polifásica de ordem $M \times M$ do banco de síntese. A Figura 3.8 ilustra a representação em polifásicas de um banco de síntese de M bandas.

Utilizando-se essa representação em polifásicas para os filtros de análise e de síntese do banco da Figura 3.6, será obtida a representação equivalente ilustrada na Figura 3.9(a). Utilizando-se identidades nobres de sistemas multitaxa [14], pode-se redesenhar o sistema na forma mostrada na Figura 3.9(b). Essa estrutura simplificada apresenta vantagem, inclusive, sob o aspecto de implementação, dado que, nessa configuração, os filtros trabalham em taxas mais baixas, pois suas entradas são sinais subamostrados. Por fim, as matrizes $\mathbf{E}(z)$ e $\mathbf{R}(z)$ podem ser combinadas em uma única matriz $\mathbf{P}(z)$, obtendo-se a representação da Figura 3.9(c), ou seja:

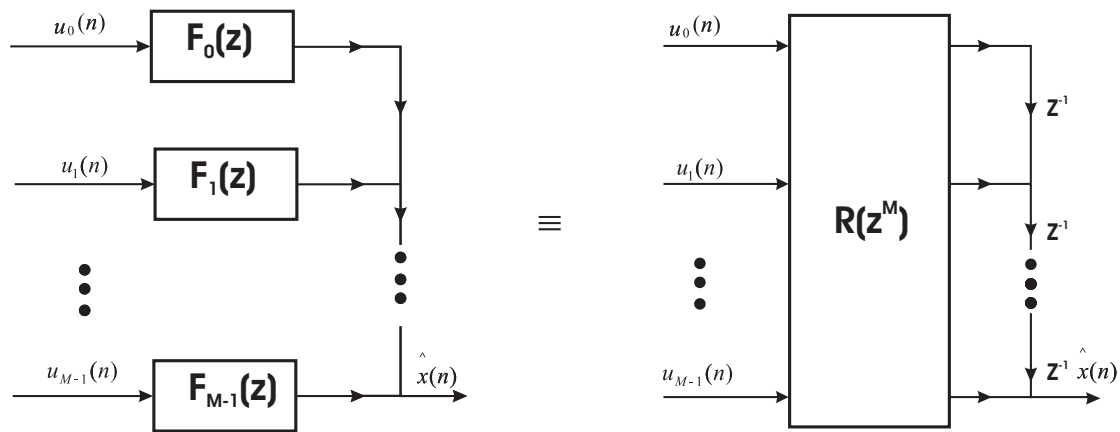


Figura 3.8: Representação polifásica de um banco de síntese com M faixas

$$\mathbf{P}(z) = \mathbf{R}(z)\mathbf{E}(z) \quad (3.46)$$

3.3.3 Reconstrução Perfeita - M Faixas

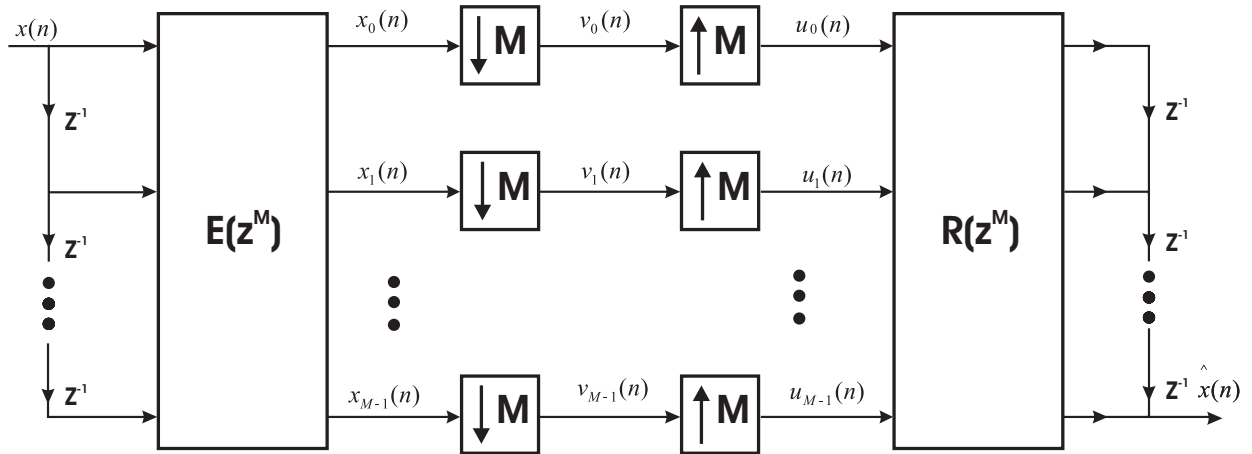
Um sistema com reconstrução perfeita é, antes de tudo, um sistema que satisfaz $\hat{x}(n) = cx(n - n_0)$. Isto é, o sinal reconstruído é uma versão escalonada e atrasada do sinal original. Isso significa que se conseguiu eliminar o efeito de *aliasing* e que, portanto, a função de transferência $T(z)$ é caracterizada por um simples atraso. Nessa seção avaliaremos sob que condições sistemas como o da Figura 3.9(b) podem apresentar reconstrução perfeita.

A discussão será iniciada baseando-se em aspectos intuitivos que serão, posteriormente, generalizados. Considerando-se o sistema ilustrado na Figura 3.9(b), pode-se notar que, fazendo-se:

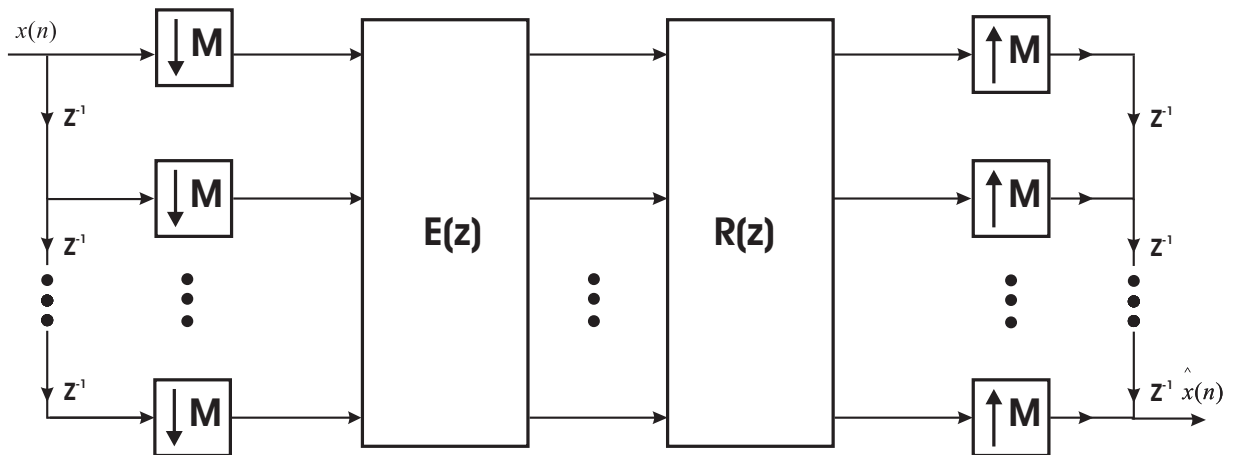
$$\mathbf{E}(z)\mathbf{R}(z) = \mathbf{I}, \quad (3.47)$$

a saída $\hat{x}(n)$ não será modificada, garantindo a desejada característica de reconstrução perfeita. Em seguida, as matrizes podem ser rearranjadas para se obter o sistema equivalente da Figura 3.9(a), que continuará, portanto, com reconstrução perfeita.

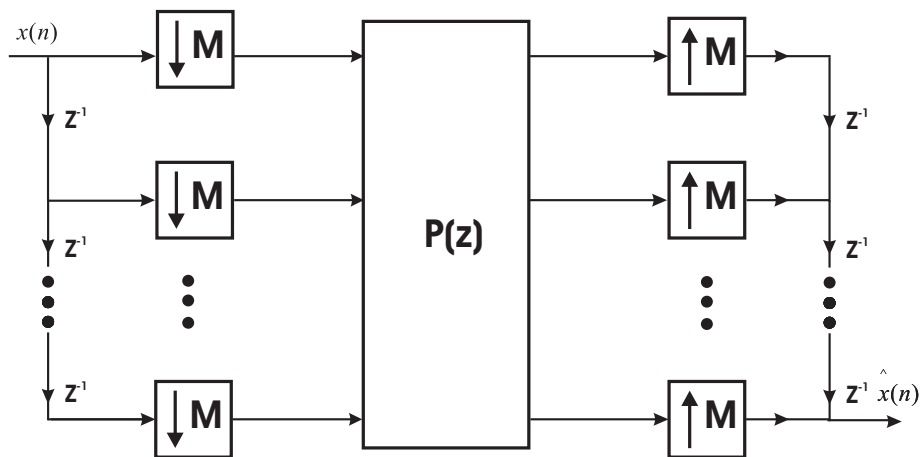
Agora pode-se então pensar de uma outra maneira: suponha que sejam dados os filtros de análise $H_k(z)$, $0 \leq k \leq M - 1$. Assim, a matriz $\mathbf{E}(z)$ já estará completamente determinada. Assumindo-se que $\mathbf{E}(z)$ possa ser invertida, poder-se-ia obter um sistema com reconstrução perfeita pela simples escolha $\mathbf{R}(z) = \mathbf{E}^{-1}(z)$. Finalmente, os filtros de síntese seriam obtidos pelas suas componentes polifásicas já determinadas. No entanto, surgem novas questões a respeito da inversibilidade da matriz polinomial $\mathbf{E}(z)$. Felizmente, a inversão direta dessa matriz pode ser evitada de diferentes maneiras. Uma dessas maneiras é restringi-la a ser



(a)



(b)



(c)

Figura 3.9: Banco de Filtros Digitais: (a) Representação Polifásica de um banco de filtros com M faixas criticamente decimado; (b) Rearranjo a partir de identidades nobres; (c) Simplificação, em que $\mathbf{P}(z) = \mathbf{R}(z)\mathbf{E}(z)$.

paraunitária. Nesse caso, tem-se:

$$\tilde{\mathbf{E}}(z)\mathbf{E}(z) = d\mathbf{I}, \text{ isto é, } \mathbf{E}^{-1}(z) = \tilde{\mathbf{E}}(z)/d, \text{ para todo } z, \text{ e } d > 0. \quad (3.48)$$

Nesse caso especial, a inversa de $\mathbf{E}(z)$ é a sua matriz paraconjugada $\tilde{\mathbf{E}}(z)$ [10].

A condição (3.47), suficiente para que se obtenha reconstrução perfeita, pode ser generalizada. Assim, ainda se terá reconstrução perfeita com:

$$\mathbf{E}(z)\mathbf{R}(z) = cz^{-m_0}\mathbf{I}, \quad (3.49)$$

para algum algum inteiro $0 \leq r \leq M - 1$, qualquer inteiro m_0 , e para uma constante $c \neq 0$. Nesse caso, será associada uma função de transferência com atraso, $T(z) = cz^{-(Mm_0+M-1)}$. Nessas condições, o sinal reconstruído será uma versão atrasada do sinal de entrada, isto é, $\hat{x}(n) = cx(n - n_0)$, com $n_0 = Mm_0 + r + M - 1$.

Pode-se generalizar um pouco mais esses resultados observando-se que um sistema com reconstrução perfeita nada mais é que um sistema sem efeito de *aliasing* com função de transferência $T(z) = \text{atraso}$, como já mencionado. Para se assegurar que o efeito de *aliasing* seja eliminado, basta se impor que a matriz $\mathbf{P}(z)$ seja pseudocirculante. O próximo tópico explicará esse conceito de maneira breve.

Matrizes Pseudocirculantes

Uma matriz é dita circulante quando cada uma de suas linhas é obtida pelo deslocamento em uma posição de cada elemento da linha anterior, lembrando-se que o elemento mais à direita quando deslocado deve aparecer na primeira coluna da linha seguinte. Um exemplo de matriz circulante é:

$$\begin{bmatrix} P_0(z) & P_1(z) & P_2(z) \\ P_2(z) & P_0(z) & P_1(z) \\ P_1(z) & P_2(z) & P_0(z) \end{bmatrix}. \quad (3.50)$$

Na verdade, talvez fosse mais adequado chamar a matriz circulante de matriz circulante à direita, visto que apenas deslocamentos nessa direção são admitidos em sua conceituação. Já a matriz pseudocirculante é essencialmente uma matriz circulante, com a característica adicional de que os elementos situados abaixo da diagonal principal devem ser multiplicados por z^{-1} . Um simples exemplo:

$$\begin{bmatrix} P_0(z) & P_1(z) & P_2(z) \\ z^{-1}P_2(z) & P_0(z) & P_1(z) \\ z^{-1}P_1(z) & z^{-1}P_2(z) & P_0(z) \end{bmatrix}. \quad (3.51)$$

A partir do exposto, vê-se que todos os elementos de uma matriz pseuocirculante podem ser determinados pela sua primeira linha. Considere então uma matriz $\mathbf{P}_{M \times M}(z)$ pseudocirculante, cuja primeira linha seja dada por:

$$\left[P_0(z) \quad P_1(z) \quad \dots \quad P_{M-1}(z) \right] \quad (3.52)$$

Sabe-se que a condição de a matriz polifásica ser pseudocirculante é necessária e suficiente para que o banco de filtros com M faixas criticamente decimado seja livre do efeito de *aliasing*. Satisfazendo-se essa condição, portanto, tem-se $\hat{X}(z) = T(z)X(z)$, e a função de transferência $T(z)$ pode ser expressa como:

$$T(z) = z^{-(M-1)}(P_0(z^M) + z^{-1}P_1(z^M) + \dots + z^{-(M-1)}P_{M-1}(z^M)), \quad (3.53)$$

em que $P_m(z)$ são os elementos da primeira linha da matriz $\mathbf{P}(z)$.

Como dito, para que um sistema tenha reconstrução perfeita, é necessário e suficiente que acumule as características de ser livre de efeito de *aliasing* e de ter função de transferência igual a um atraso. A primeira questão já se sabe que é possível garantir fazendo-se $\mathbf{P}(z)$ pseudocirculante, o que resulta na função de transferência de 3.53. Basta, então, impor condições para que essa função de transferência se torne apenas um atraso. Considerando-se que já se impôs a restrição de que a matriz seja pseudocirculante, apenas um atraso poderá ser obtido se $P_m(z) = 0$, exceto para um único valor de m no intervalo $0 \leq m \leq M - 1$. Além disso, o termo não-nulo deve ter a forma cz^{-m_0} . Resumindo, um sistema sem efeito de *aliasing* e com reconstrução perfeita deve ter a primeira linha da matriz pseudocirculante $\mathbf{P}(z)$ igual a:

$$\left[0 \quad \dots \quad 0 \quad cz^{-m_0} \quad 0 \quad \dots \quad 0 \right]. \quad (3.54)$$

De forma equivalente, $\mathbf{P}(z)$ terá a seguinte forma:

$$\mathbf{P}(z) = \mathbf{R}(z)\mathbf{E}(z) = cz^{-m_0} \begin{bmatrix} \mathbf{0} & \mathbf{I}_{M-r} \\ z^{-1}\mathbf{I}_r & \mathbf{0} \end{bmatrix}, \quad (3.55)$$

para algum r no intervalo $0 \leq r \leq M - 1$. Com essa última restrição, a função de transferência da equação 3.53, se torna:

$$T(z) = cz^{-r} z^{-(M-1)} z^{-m_0 M}. \quad (3.56)$$

Nessas condições, nota-se que o sinal reconstruído será $\hat{x}(n) = cx(n - n_0)$, em que $n_0 = Mm_0 + r + M - 1$. Pode-se perceber que satisfazer a equação 3.55, equivale a:

$$\det \mathbf{R}(z) \det \mathbf{E}(z) = c_0 z^{-k_0}, \quad (3.57)$$

para $c_0 \neq 0$ e para k_0 inteiro. Logo, qualquer sistema com reconstrução perfeita obedece a essa restrição do determinante.

Bancos de filtros com reconstrução perfeita, compostos por filtros FIR $H_k(z)$ e $F_k(z)$, são de grande interesse prático. Nesses sistemas, as matrizes polifásicas $\mathbf{E}(z)$ e $\mathbf{R}(z)$ serão também FIR, visto que seus elementos são os componentes polifásicos dos filtros. Essa natureza FIR das matrizes implica determinantes igualmente FIR. Se o produto dessas duas matrizes tem de ser um atraso, como evidenciado em (3.57), então o determinante de cada uma delas também assim deve ser. Logo:

$$\det \mathbf{E}(z) = \alpha z^{-K}, \quad \alpha \neq 0, \quad K = \text{inteiro}. \quad (3.58)$$

$\mathbf{R}(z)$ também deve ter forma similar para que todo o sistema FIR tenha reconstrução perfeita. Torna-se então importante para o projeto de um banco de filtros dessa natureza estudar que tipos de matrizes são capazes de garantir 3.58. Um tipo particular de matrizes FIR causais chamadas paraunitárias satisfazem 3.58 com K igual ao grau de McMillan [10] de $\mathbf{E}(z)$. Também existe uma outra família de matrizes chamadas unimodulares que, por definição, satisfazem 3.58 com $K = 0$. De maneira geral, foi mostrado em [18] que qualquer matriz causal que satisfaça 3.58 pode ser decomposta em um produto de uma matriz paraunitária e de uma matriz unimodular. Isso motivou grande estudo sobre esses tipos especiais de matrizes, que pode ser encontrado de forma detalhada em [10].

3.4 Bancos de Filtros Multidimensionais

3.4.1 Considerações iniciais

Enquanto muitas pesquisas a respeito de bancos de filtros unidimensionais (1-D) têm sido feitas, bem como sobre a construção das bases a esses relacionadas, existem limitados resultados sobre o caso multidimensional (M-D). Nesse caso, a prática mais frequente tem sido a de generalizar sistemas (1-D) para maiores dimensões de uma maneira separável. Por outro lado, vários estudos [19][9][8] indicam que bancos de filtros não separáveis interessantes podem ser construídos em (M-D). Porém, vale lembrar que o projeto dos filtros (M-D) não separáveis é muito mais complexo, como preliminarmente discutido no Capítulo 2.

Recentemente, tem existido grande esforço na busca por uma “verdadeira” representação

bidimensional de imagens [20]. Um dos maiores espaços a serem preenchidos no que já foi desenvolvido para o processamento de imagens é sobre como se pode explorar a estrutura geométrica intrínseca a esses sinais. Dentre desenvolvimentos atuais em representações (2-D) multiescalonáveis estão as construções *ridgelet* e *curvelet* [21], [22], e seus resultados indicam que o aproveitamento da direcionalidade é crucial para a representação eficiente de uma imagem. Isso tanto é suportado por estudos mais antigos como os realizados sobre o sistema visual humano [23], tanto por estudos mais recentes sobre identificação de componentes esparsos em imagens naturais [24].

Na literatura sobre bancos de filtros, Bamberger e Smith ainda em 1992 [25] já tinham desenvolvido um banco de filtros (2-D) direcional (DFB - do inglês, “Directional Filter Bank”), que poderia ser criticamente decimado, permitindo reconstrução perfeita.

3.4.2 Conceitos básicos sobre sinais multidimensionais

Um sinal D-dimensional $x_a(t_0, t_1, \dots, t_{D-1})$ é uma função de D variáveis reais t_0, t_1, \dots, t_{D-1} . Pode-se, então, definir o seguinte vetor coluna:

$$\mathbf{t} = \begin{bmatrix} t_0 & t_1 & \dots & t_{D-1} \end{bmatrix}^T, \quad (3.59)$$

como também se pode abreviar a escrita do sinal para $\{x_a(\mathbf{t})\}$. O subscrito a indica que o sinal é analógico, porque as variáveis t_i foram consideradas contínuas. Analogamente, se o sinal for discreto, pode-se definir outro vetor coluna:

$$\mathbf{n} = \begin{bmatrix} n_0 & n_1 & \dots & n_{D-1} \end{bmatrix}^T. \quad (3.60)$$

Nesse caso, escreve-se o sinal $\{x(\mathbf{n})\}$ é uma função de D variáveis inteiras $n_k, 0 \leq k \leq D - 1$, ou seja, dos elementos do vetor \mathbf{n} . A transformada de Fourier, se existir, é definida por analogia ao caso (1-D), e é função de D variáveis de frequência $\omega_k, 0 \leq k \leq D - 1$. Considerando-se:

$$\boldsymbol{\omega} = \begin{bmatrix} \omega_0 & \omega_1 & \dots & \omega_{D-1} \end{bmatrix}^T, \quad (3.61)$$

a transformada de Fourier é definida por:

$$X(\boldsymbol{\omega}) = \sum_{\mathbf{n} \in \mathcal{N}} x(\mathbf{n}) e^{-j\boldsymbol{\omega}^T \mathbf{n}}, \quad (3.62)$$

em que \mathcal{N} denota o conjunto de todos os vetores com D componentes inteiros.

Pode-se também definir a transformada z do sinal $\{\mathbf{x}(\mathbf{n})\}$:

$$X(\mathbf{z}) = \sum_{\mathbf{n} \in \mathcal{N}} x(\mathbf{n}) z_0^{-n_0} z_1^{-n_1} \dots z_{D-1}^{-n_{D-1}}, \quad (3.63)$$

em que \mathbf{z} denota o vetor:

$$\mathbf{z} = \left[z_0 z_1 \dots z_{D-1} \right]^T. \quad (3.64)$$

Introduzindo a notação:

$$\mathcal{Z}(\mathbf{n}) \triangleq z_0^{n_0} z_1^{n_1} \dots z_{D-1}^{n_{D-1}}, \quad (3.65)$$

a transformada z se torna:

$$X(\mathbf{z}) = \sum_{\mathbf{n} \in \mathcal{N}} x(\mathbf{n}) \mathcal{Z}(-\mathbf{n}). \quad (3.66)$$

Essa notação será utilizada na definição dos expansores multidimensionais. A partir do exposto, nota-se que, para o caso especial $D = 2$, a seguinte notação pode ser utilizada: $\{x(n_0, n_1)\}$, para representar o sinal digital, e $X_z(z_0, z_1)$, sua transformada z .

3.4.3 Filtros Digitais Multidimensionais

Um filtro digital multidimensional é um sistema linear invariante ao deslocamento caracterizado pela seguinte função de transferência:

$$H(\mathbf{z}) = \sum_{\mathbf{n} \in \mathcal{N}} h(\mathbf{n}) \mathcal{Z}(-\mathbf{n}), \quad (3.67)$$

em que $h(\mathbf{n})$ é a resposta impulsional do filtro. Assim, a saída de um sistema com essa resposta impulsional poderá ser obtida da seguinte forma, em função das transformadas z :

$$Y(\mathbf{z}) = H(\mathbf{z})X(\mathbf{z}), \quad (3.68)$$

que equivale à seguinte operação de convolução:

$$y(\mathbf{n}) = \sum_{\mathbf{k} \in \mathcal{N}} x(\mathbf{k}) h(\mathbf{n} - \mathbf{k}) = \sum_{\mathbf{k} \in \mathcal{N}} h(\mathbf{k}) x(\mathbf{n} - \mathbf{k}). \quad (3.69)$$

É importante observar que, no caso de sinais $\{x(\mathbf{n})\}$ finitos, como uma imagem, a operação de convolução em (3.69) resultará em um sinal $\{y(\mathbf{n})\}$ de tamanho maior que o do sinal original. No entanto, várias técnicas para reduzir o tamanho de $\{y(\mathbf{n})\}$ já são conhecidas promovendo o adequado tratamento das bordas.

A resposta em frequência $H(\boldsymbol{\omega})$ do filtro é dada por:

$$H(\boldsymbol{\omega}) = \sum_{\mathbf{n} \in \mathcal{N}} h(\mathbf{n}) e^{-j \boldsymbol{\omega}^T \mathbf{n}}. \quad (3.70)$$

Essa resposta é $H(\mathbf{z})$ avaliada em $z_k = e^{j\omega_k}$, $0 \leq k \leq D - 1$. Já no caso de filtros separáveis, (3.70) deve ser reescrita como:

$$H(\boldsymbol{\omega}) = H_0(e^{j\omega_0}) H_1(e^{j\omega_1}) \dots H_{D-1}(e^{-j\omega_{D-1}}), \quad (3.71)$$

isto é, a resposta multifrequencial é o produto de funções de uma única frequência.

A vantagem dos filtros separáveis é que esses podem ser facilmente obtidos a partir de filtros unidimensionais. No entanto, também existem filtros não separáveis que podem ser obtidos a partir de versões 1-D por mapeamentos simples. Um exemplo é um filtro com resposta em frequência $1 - H(\boldsymbol{\omega})$, que não é necessariamente separável, ainda que $H(\boldsymbol{\omega})$ seja.

3.4.4 Amostragem de sinais Multidimensionais

Como mencionado no Capítulo 2, uma das maneiras de se obter um sistema não separável é realizar uma amostragem não separável, como, por exemplo, a amostragem em *quincunx* [10]. Diferentemente do caso 1-D, em que existe apenas uma maneira de se amostrar o sinal, a base teórica da amostragem de sinais multidimensionais é fundamentalmente mais complicada porque existem inúmeras maneiras de se definir a geometria de amostragem. Assim, em um sistema multitaxa multidimensional, operações de amostragem são definidas em reticulados (do inglês, *Lattice*). Um reticulado \mathcal{L} , em D dimensões é representada por uma matriz $D \times D$ não-singular \mathbf{M} como:

$$\mathcal{L}(\mathbf{M}) = \{\mathbf{m} : \mathbf{m} = \mathbf{M}\mathbf{n}, n \in \mathcal{Z}^D\}, \quad (3.72)$$

em outras palavras, as coordenadas dos pontos pertencentes ao reticulado $\mathcal{L}(\mathbf{M})$ são obtidas por combinações lineares das colunas de \mathbf{M} . Apenas para ilustrar, considere o caso 2-D. Dado um sinal dessa natureza, $x_a(t_0, t_1)$, o jeito mais imediato de se amostrá-lo seria:

$$x(n_0, n_1) = x_a(n_0 T_0, n_1 T_1), \quad (3.73)$$

em que n_0 e n_1 são inteiros.

Os pontos amostrados podem então ser arranjados em um padrão retangular, o que confere a esse método o termo amostragem retangular. Em geral, pode-se notar que o caso retangular é representado por matrizes diagonais como:

$$\mathbf{M} = \begin{bmatrix} T_0 & 0 \\ 0 & T_1 \end{bmatrix}. \quad (3.74)$$

Deve-se ressaltar, porém, a importância em se apresentar padrões mais gerais de amostragem, visto que a amostragem retangular frequentemente não é o jeito mais eficiente de se amostrar um sinal passa-faixa. Antes que se proceda à generalização, será dado um outro exemplo ilustrativo, dessa vez para que se possa entender como podem ser obtidas as coordenadas dos pontos amostrados por meio dos vetores da matriz de amostragem.

Considere a sequência gerada por:

$$x(n_0, n_1) = x_a(n_0 - n_1, n_0 + 2n_1). \quad (3.75)$$

Os pontos de amostragem são mostrados na Figura 3.10. Para entender melhor como os pontos foram obtidos, basta se notar que as localizações das amostras, composta pelas coordenadas $t_0 = n_0 - n_1$ e $t_1 = n_0 + 2n_1$, podem ser escritas na seguinte forma compacta:

$$\mathbf{t} \triangleq \begin{bmatrix} t_0 \\ t_1 \end{bmatrix} = \underbrace{\begin{bmatrix} 1 & -1 \\ 1 & 2 \end{bmatrix}}_{\mathbf{M}} \begin{bmatrix} n_0 \\ n_1 \end{bmatrix}. \quad (3.76)$$

Cada localização \mathbf{t} da amostra pode, então, ser escrita na forma:

$$\mathbf{t} = n_0 \mathbf{m}_0 + n_1 \mathbf{m}_1, \quad (3.77)$$

em que $\mathbf{m}_0 = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$ e $\mathbf{m}_1 = \begin{bmatrix} -1 \\ 2 \end{bmatrix}$.

Dessa maneira, dada a matriz \mathbf{M} , os pontos amostrados podem ser obtidos traçando-se, inicialmente, os vetores dessa matriz e, em seguida, retas paralelas a esses vetores, formando paralelogramos. Os pontos estarão localizados onde as retas se cruzam, como ilustrado na Figura 3.10.

Apenas para ilustrar, no caso da amostragem *quincunx*, cujo conjunto de amostras é mostrado na Figura 3.11, a matriz de amostragem \mathbf{Q} seria dada por:

$$\mathbf{Q} = \begin{bmatrix} 1 & 1 \\ -1 & 1 \end{bmatrix}. \quad (3.78)$$

Como visto no exemplo, a amostragem de um sinal D-dimensional pode ser representada em função da matriz de amostragem:

$$x(\mathbf{n}) = x_a(\mathbf{M}\mathbf{n}), \quad (3.79)$$

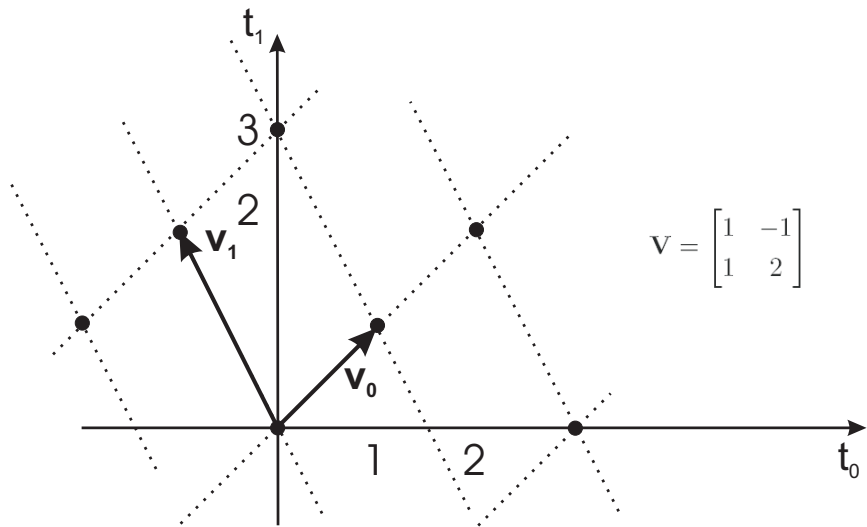


Figura 3.10: Conjunto de pontos resultante da amostragem $x(n_0, n_1) = x_a(n_0 - n_1, n_0 + 2n_1)$

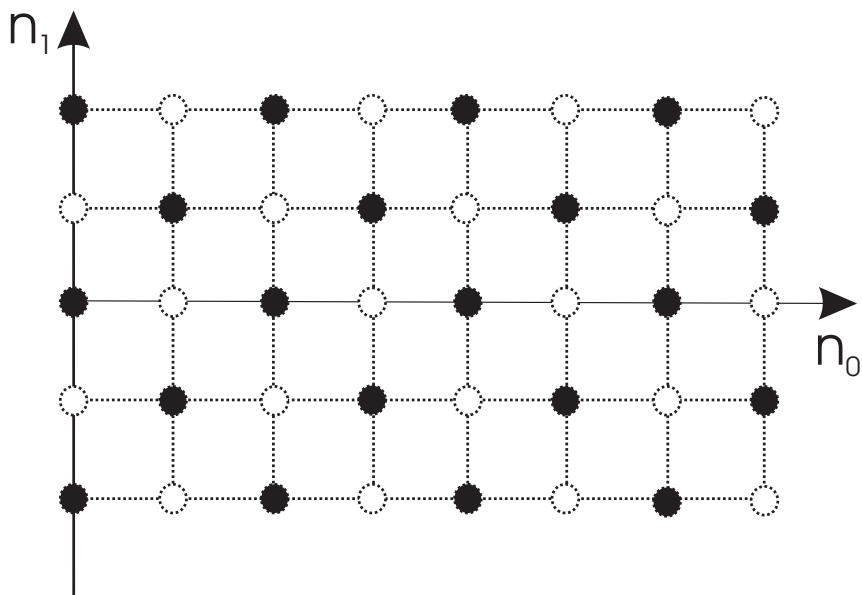


Figura 3.11: Conjunto de amostras *quincunx*.

em que $\mathbf{M}_{D \times D}$ é uma matriz não-singular, contendo os D vetores em suas colunas:

$$\mathbf{M} = \left[\mathbf{m}_0, \mathbf{m}_1, \dots, \mathbf{m}_{D-1} \right] \quad (3.80)$$

e $\mathbf{n} \in \mathcal{N}$. O conjunto de todas as amostras é:

$$\mathbf{t} = \mathbf{M}\mathbf{n}, \quad \mathbf{n} \in \mathcal{N}, \quad (3.81)$$

isto é, o conjunto dos vetores $\sum_{k=0}^{D-1} n_k \mathbf{m}_k$. Esse é o conjunto de todas as combinações lineares inteiras das colunas $\mathbf{m}_0, \dots, \mathbf{m}_{D-1}$ de \mathbf{M} . A matriz \mathbf{M} tanto é chamada matriz de amostragem quanto de base geradora do reticulado $LAT(\mathbf{M})$. Note que o conjunto $LAT(\mathbf{M})$ é discreto, possuindo, porém, infinitos elementos.

Após a apresentação da matriz de amostragem, torna-se oportuno definir o vetor coluna $\mathbf{z}^{(\mathbf{M})}$, obtido a partir do vetor de coluna $\mathbf{z} = [z_0, \dots, z_{D-1}]^T$.

Pode-se então definir $\mathbf{z}^{(\mathbf{M})}$, que é um vetor coluna com D componentes obtido a partir do vetor $\mathbf{z} = [z_0, \dots, z_{D-1}]^T$. O k -ésimo elemento de $\mathbf{z}^{(\mathbf{M})}$ é dado por:

$$(\mathbf{z}^{(\mathbf{M})})_k = z_0^{M_{0,k}} z_1^{M_{1,k}} \dots z_{D-1}^{M_{D-1,k}}, \quad (3.82)$$

em que $M_{0,k}, M_{1,k}, \dots, M_{D-1,k}$, com $0 \leq k \leq D-1$ são os elementos da matriz de amostragem \mathbf{M} .

3.5 Fundamentos de sistemas multitaxa multidimensionais

De forma análoga aos sistemas unidimensionais discutidos nas seções anteriores, serão definidos, para o caso multidimensional, conceitos de decimação, interpolação, estruturas de bancos de filtros, representações polifásicas, entre outros relacionados à composição de sistemas multitaxas.

3.5.1 Componentes Básicos

Primeiramente serão definidos os operadores responsáveis por alterarem a taxa de amostragem do sinal de entrada: os decimadores, que a diminuem; e os expansores (ou sobreamostradores), que a elevam.

Dada uma sequência (1-D) $\{x(n)\}$, sua versão decimada por um fator M , foi definida como sendo $x_d(n) = x(nM)$. Analogamente, dada uma sequência D-dimensional $\{x(\mathbf{n})\}$, a relação entre esse sinal de entrada e sua versão decimada por uma matriz \mathbf{M} , $x_d(\mathbf{n})$ será:

$$\begin{aligned} x_d(\mathbf{n}) &= x(\mathbf{M}\mathbf{n}) \\ X_d(\boldsymbol{\omega}) &= \frac{1}{|\det(\mathbf{M})|} \sum_{\mathbf{k} \in \mathcal{N}(\mathbf{M}^T)} X(\mathbf{M}^{-T}\boldsymbol{\omega} - 2\pi\mathbf{M}^{-T}\mathbf{k}). \end{aligned} \quad (3.83)$$

Aqui, $\mathcal{N}(\mathbf{M})$ é definido como o conjunto dos vetores inteiros na forma $\mathbf{M}\mathbf{t}$, em que $\mathbf{t} \in [0, 1)^D$. O número de elementos $J(\mathbf{M})$ em $\mathcal{N}(\mathbf{M})$ é dado por:

$$J(\mathbf{M}) = |\det(\mathbf{M})|, \quad (3.84)$$

e a matriz $\mathbf{M}^{-T} = (\mathbf{M}^T)^{-1}$ é que gera o reticulado recíproco à $\mathcal{L}(\mathbf{M})$, consistindo, portanto, de pontos que representam o espectro replicado em frequência.

A partir do exposto na seção 3.4.4, já se sabe que existem inúmeros tipos de amostragem, não apenas a amostragem retangular. Essa generalização é bastante importante, pois, dependendo da resposta em frequência do sinal a ser subamostrado, dado um fator de decimação $J(\mathbf{M})$, é possível que o método retangular gere *aliasing*, por exemplo, enquanto uma amostragem do tipo hexagonal não [10].

Para o caso unidimensional, um expansor ou sobreamostrador foi definido como sendo:

$$x_e(n) = \begin{cases} x(n/M), & \text{n múltiplo de } M; \\ 0, & \text{demais casos.} \end{cases} \quad (3.85)$$

O expansor multidimensional será definido de maneira semelhante, porém em termos de uma matriz \mathbf{M} . Mais precisamente, a relação entre o sinal de entrada $x(\mathbf{n})$ e sua versão expandida pela matriz \mathbf{M} , $x_e(\mathbf{n})$ será:

$$\begin{aligned} x_e(\mathbf{n}) &= \begin{cases} x(\mathbf{M}^{-1}\mathbf{n}), & \text{se } \mathbf{n} \in \mathcal{L}(\mathbf{M}), \\ 0, & \text{demais casos.} \end{cases} \\ X_u(\boldsymbol{\omega}) &= X(\mathbf{M}^T\boldsymbol{\omega}), \\ X_u(\mathbf{z}) &= X(\mathbf{z}^{(\mathbf{M})}). \end{aligned} \quad (3.86)$$

Observa-se que, na equação acima, foi utilizada a simbologia explicitada em 3.82. Para que se possa entender melhor esse conceito aplicado aos expansores, será considerado um exemplo de amostragem retangular. Assim, dada a matriz de amostragem diagonal $\mathbf{M} = \begin{bmatrix} 2 & 0 \\ 0 & 3 \end{bmatrix}$, se terá $X_u(z_0, z_1) = X(z_0^2, z_1^3)$.

Existem casos especiais em que as operações de amostragem usam matrizes unimodulares inteiras, isto é, matrizes com determinante igual a ± 1 . Amostragem por matrizes unimodulares inteiras não alteram a taxa de amostragem dos dados, apenas rearranjam as amostras de entrada, o que é comumente chamado de reamostragem (ou, em inglês, *resampling*).

3.5.2 Representação Polifásica - Multidimensional

Dada uma sequência unidimensional $x(n)$, essa pode ser decomposta em seus componentes polifásicos $x_k(n) = x(nM + k)$, $0 \leq k \leq M - 1$, como mostrado na Seção 3.2.4. Pode-se então expressar $X(z) = \sum_{k=0}^{M-1} z^{-k} X_k(z^M)$, como utilizado na representação dos filtros de um banco com M bandas na seção 3.3.2. Essa decomposição se baseia no fato de que qualquer inteiro n pode ser representado da seguinte maneira:

$$n = k + Mq_0, \quad (3.87)$$

em que k e q_0 são inteiros, tais que $0 \leq k \leq M - 1$. De outra maneira, pode-se perceber que k é o resto e q_0 o quociente da divisão de n por M . Esse mesmo teorema da divisão para escalares inteiros pode ser aplicado à vetores de inteiros. Assim, seja \mathbf{M} uma matriz não-singular inteira de ordem $D \times D$, e seja \mathbf{n} um inteiro D -dimensional. Pode-se expressar \mathbf{n} como:

$$\mathbf{n} = \mathbf{k} + \mathbf{M}\mathbf{q}_0, \quad (3.88)$$

em que $\mathbf{k} \in \mathcal{N}(\mathbf{M})$ e \mathbf{q}_0 é um vetor inteiro. Assim, dada uma sequência $x(\mathbf{n})$ e uma matriz \mathbf{M} , podem ser identificadas $J(\mathbf{M})$ subsequências únicas, tais que:

$$x_{\mathbf{k}}(\mathbf{n}) = x(\mathbf{M}\mathbf{n} + \mathbf{k}), \quad \mathbf{k} \in \mathcal{N}(\mathbf{M}). \quad (3.89)$$

Os sinais $\{x_{\mathbf{k}}(\mathbf{n})\}$ são os componentes polifásicos de $\{x(\mathbf{n})\}$. Note que $x_{\mathbf{k}}(\mathbf{n})$ é, simplesmente, a versão decimada de $x(\mathbf{n} + \mathbf{k})$.

Definindo-se $y_{\mathbf{k}}(\mathbf{n})$ como a versão \mathbf{M} -expandida de $x_{\mathbf{k}}(\mathbf{n})$:

$$y_{\mathbf{k}}(\mathbf{n}) = \begin{cases} x_{\mathbf{k}}(\mathbf{M}^{-1}\mathbf{n}), & \mathbf{n} \in LAT(\mathbf{M}) \\ 0, & \text{demais casos,} \end{cases} \quad (3.90)$$

pode-se expressar cada amostra $x(\mathbf{n})$ em termos de um dos $y_{\mathbf{k}}(\mathbf{n})$. Em outras palavras, dada uma amostra de $x(\mathbf{n})$, sempre se pode achar um, e apenas um, componente polifásico do sinal para representá-la. Para tal, deve-se calcular $\mathbf{k} = \mathbf{n} \bmod \mathbf{M}$, isto é, o único inteiro em $\mathcal{N}(\mathbf{M})$ que satisfaz 3.89, para, então, se obter:

$$x(\mathbf{n}) = y_{\mathbf{k}}(\mathbf{n} - \mathbf{k}). \quad (3.91)$$

Considere a seguinte matriz de amostragem, correspondente a um padrão hexagonal:

$$\mathbf{M} = \begin{bmatrix} 1 & 1 \\ 2 & -2 \end{bmatrix}. \quad (3.92)$$

Deseja-se encontrar os componentes polifásicos relacionados a esse tipo de amostragem. Por inspeção, sabe-se que o conjunto $\mathcal{N}(\mathbf{M})$ terá 4 elementos, visto que $J(\mathbf{M}) = |\det(\mathbf{M})| = |-4| = 4$. Sabe-se, também, que esses quatro elementos são vetores inteiros pertencentes ao que se chama paralelepípedo fundamental (*FPD*, do inglês *Fundamental Parallelepiped*). De maneira simples, o *FPD* pode ser obtido traçando-se os vetores de base \mathbf{m}_k , isto é, as colunas da matriz \mathbf{M} e, em seguida completando-se o paralelepípedo usando os vetores como arestas. A Figura 3.12 mostra o *FPD*(\mathbf{M}) e os quatro vetores dentro dele. O conjunto $\mathcal{N}(\mathbf{M})$ é dado por:

$$\mathbf{k}_0 = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \quad \mathbf{k}_1 = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \quad \mathbf{k}_2 = \begin{bmatrix} 1 \\ 1 \end{bmatrix}, \quad \mathbf{k}_3 = \begin{bmatrix} 1 \\ -1 \end{bmatrix}. \quad (3.93)$$

Por definição, os vetores \mathbf{k}_i possuem a forma $\mathbf{M}\mathbf{x}_i$, com $\mathbf{x}_i \in [0, 1)^2$. Assim, a partir dos vetores \mathbf{k}_i e da matriz \mathbf{M}^{-1} podem ser obtidos os componentes polifásicos \mathbf{x}_i :

$$\mathbf{x}_0 = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \quad \mathbf{x}_1 = \begin{bmatrix} 0.5 \\ 0.5 \end{bmatrix}, \quad \mathbf{x}_2 = \begin{bmatrix} 0.75 \\ 0.25 \end{bmatrix}, \quad \mathbf{x}_3 = \begin{bmatrix} 0.25 \\ 0.75 \end{bmatrix}. \quad (3.94)$$

No entanto, também é desejável escrever a decomposição polifásica multidimensional no domínio z para obter uma expressão análoga a (3.23). Lembrando que o k -ésimo componente polifásico, $x_{\mathbf{k}}(\mathbf{n})$, de $x(\mathbf{n})$ foi definido em (3.89), e que $x(\mathbf{n})$ pode ser escrito unicamente em termos de componentes $y_{\mathbf{k}}(\mathbf{n})$ como em (3.91), obtêm-se as relações mostradas, em termos de diagramas de blocos no domínio z , na Figura 3.13. Combinando-se essas relações, chega-se à Figura 3.14, que meramente mostra a decomposição de $x(\mathbf{n})$ em seus componentes polifásicos, e a sua recomposição a partir desses.

A partir dos diagramas, percebe-se que:

$$X(\mathbf{z}) = \sum_{\mathbf{k} \in \mathcal{N}(\mathbf{M})} \mathcal{Z}(-\mathbf{k}) X_{z, \mathbf{k}}(\mathbf{z}^{(\mathbf{M})}). \quad (3.95)$$

Assim, para a seguinte matriz de decimação retangular:

$$\mathbf{M} = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix}.$$

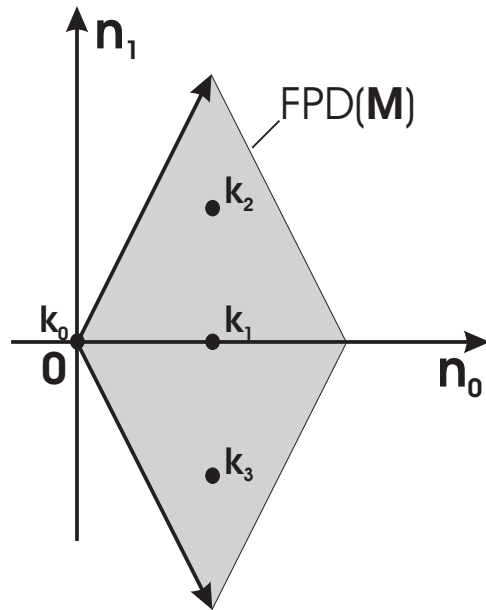


Figura 3.12: Demonstração dos quatro componentes polifásicos gerados para amostragem do tipo hexagonal, localizados no interior do $FPD(M)$.

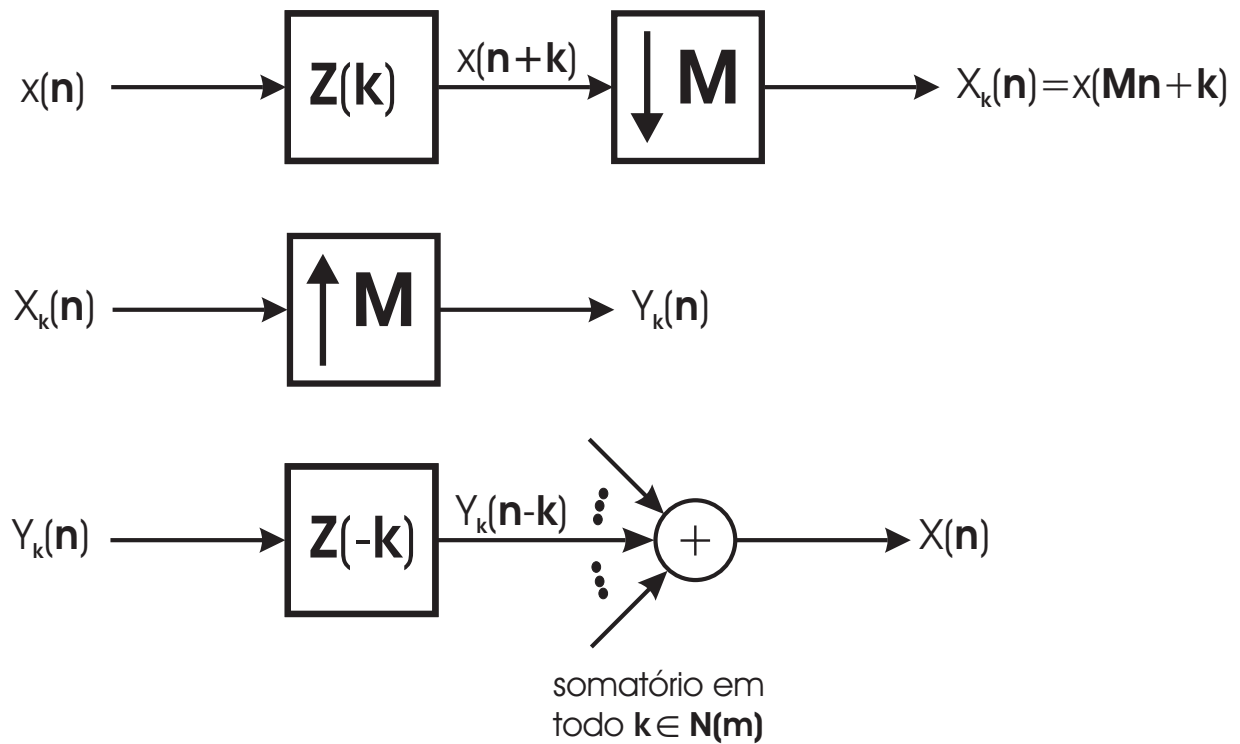


Figura 3.13: Representação em diagrama de blocos das relações entre os sinais $x(\mathbf{n})$, $x_k(\mathbf{n})$ e $y_k(\mathbf{n})$.

Tem-se então $J(\mathbf{M}) = 4$, e os vetores inteiros em $\mathcal{N}(\mathbf{M})$ são:

$$\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \end{bmatrix}, e \begin{bmatrix} 1 \\ 1 \end{bmatrix}.$$

Assim, as variáveis $\mathcal{Z}(-\mathbf{k})$ em (3.95) serão:

$$\mathcal{Z}(-\mathbf{k}_0) = 1, \mathcal{Z}(-\mathbf{k}_1) = z_0^{-1}, \mathcal{Z}(-\mathbf{k}_2) = z_1^{-1}, \mathcal{Z}(-\mathbf{k}_3) = z_0^{-1}z_1^{-1}. \quad (3.98)$$

Tem-se também que $\mathbf{z}^{(\mathbf{M})} = \begin{bmatrix} z_0^2 \\ z_1^2 \end{bmatrix}$, de tal forma que (3.95) se torna:

$$X_z(z_0, z_1) = X_{z, \mathbf{k}_0}(z_0^2, z_1^2) + z_0^{-1}X_{z, \mathbf{k}_1}(z_0^2, z_1^2) + z_1^{-1}X_{z, \mathbf{k}_2}(z_0^2, z_1^2) + z_0^{-1}z_1^{-1}X_{z, \mathbf{k}_3}(z_0^2, z_1^2). \quad (3.99)$$

Agora, considere que $X_z(\mathbf{z})$ seja a transformada z de um filtro bidimensional componente de um banco com D bandas (ou D filtros). Cada um desses filtros poderia ser decomposto em seus componentes polifásicos como em (3.100). Apenas se reescreverá essa equação para que se possa fazer referência a cada um dos filtros $E_i(z_0, z_1)$, com $i = 0, \dots, D - 1$:

$$E_i(z_0, z_1) = E_{i, \mathbf{k}_0}(z_0^2, z_1^2) + z_0^{-1}E_{i, \mathbf{k}_1}(z_0^2, z_1^2) + z_1^{-1}E_{i, \mathbf{k}_2}(z_0^2, z_1^2) + z_0^{-1}z_1^{-1}E_{i, \mathbf{k}_3}(z_0^2, z_1^2) \quad (3.100)$$

Assim, de maneira análoga ao caso unidimensional mostrado em 3.41, pode-se montar a matriz polifásica referente a esse banco de filtros:

$$\mathbf{E}(\mathbf{z}) = \begin{bmatrix} E_{0, \mathbf{k}_0}(z_0, z_1) & E_{0, \mathbf{k}_1}(z_0, z_1) & E_{0, \mathbf{k}_2}(z_0, z_1) & E_{0, \mathbf{k}_3}(z_0, z_1) \\ E_{1, \mathbf{k}_0}(z_0, z_1) & E_{1, \mathbf{k}_1}(z_0, z_1) & E_{1, \mathbf{k}_2}(z_0, z_1) & E_{1, \mathbf{k}_3}(z_0, z_1) \\ E_{2, \mathbf{k}_0}(z_0, z_1) & E_{2, \mathbf{k}_1}(z_0, z_1) & E_{2, \mathbf{k}_2}(z_0, z_1) & E_{2, \mathbf{k}_3}(z_0, z_1) \\ E_{3, \mathbf{k}_0}(z_0, z_1) & E_{3, \mathbf{k}_1}(z_0, z_1) & E_{3, \mathbf{k}_2}(z_0, z_1) & E_{3, \mathbf{k}_3}(z_0, z_1) \end{bmatrix} \quad (3.101)$$

Pode-se notar que a diferença fundamental entre a matriz polifásica multidimensional acima e aquela mostrada em (3.41) consiste no fato de que cada um dos elementos em (3.101) é uma função de z_0 e de z_1 , enquanto, claramente, no caso unidimensional tem-se apenas z .

Considerando toda a notação desenvolvida para o caso multidimensional e o exemplo acima, pode-se então generalizar a composição da matriz polifásica de um banco multidimensional com D filtros para:

$$\mathbf{E}(\mathbf{z}) = \begin{bmatrix} E_{0,\mathbf{k}_0}(\mathbf{z}) & E_{0,\mathbf{k}_1}(\mathbf{z}) & \cdots & E_{0,\mathbf{k}_{J(\mathbf{M})-1}}(\mathbf{z}) \\ E_{1,\mathbf{k}_0}(\mathbf{z}) & E_{1,\mathbf{k}_1}(\mathbf{z}) & \cdots & E_{1,\mathbf{k}_{J(\mathbf{M})-1}}(\mathbf{z}) \\ \vdots & \vdots & \ddots & \vdots \\ E_{D-1,\mathbf{k}_0}(\mathbf{z}) & E_{D-1,\mathbf{k}_1}(\mathbf{z}) & \cdots & E_{D-1,\mathbf{k}_{J(\mathbf{M})-1}}(\mathbf{z}) \end{bmatrix} \quad (3.102)$$

em que \mathbf{M} é a matriz de amostragem; $J(\mathbf{M})$ é a quantidade de polifásicas, definida em (3.84); e $\mathbf{k}_0, \dots, \mathbf{k}_{J(\mathbf{M})-1}$ é o conjunto de elementos em $\mathcal{N}(\mathbf{M})$.

Esse exemplo é parte fundamental para o entendimento da montagem da matriz polifásica de uma transformada bidimensional não-separável. Esse também é o cerne deste trabalho, porque a análise da estabilidade numérica recai sobre essa importante matriz.

3.6 Transformadas *wavelet* e bancos de filtros

3.6.1 Histórico

Bancos de filtros e transformadas *wavelet* são novas maneiras de ver e representar um sinal, alternativamente à análise de Fourier. Apesar de as transformadas *wavelet* terem se tornado muito populares e amplamente utilizadas como ferramenta de análise e de processamento de sinais recentemente, na década de 80, as bases para sua concepção já se encontravam escondidas em trabalhos de matemáticos há mais de um século. Em 1873, Karl Weierstrass descreveu como uma família de funções poderia ser obtida a partir da superposição de versões escalonadas de uma dada função de base. Matematicamente, uma onda é representada por uma função oscilante no tempo ou no espaço. A análise de Fourier expande um sinal arbitrário em infinitas funções senoidais e suas harmônicas, e tem sido muito estudada pela comunidade de processamento de sinais por décadas. É sabido que a representação de Fourier é bastante eficiente na análise de sinais periódicos e invariantes no tempo. Diferentemente da onda senoidal longa, utilizada na representação de Fourier, a *wavelet* é uma onda pequena, cuja energia é concentrada no tempo.

O termo *wavelet* foi originalmente utilizado no campo de estudo sismológico para descrever os distúrbios decorrentes de um impulso sísmico. Em 1982, Morlet et al. primeiramente descreveu como *wavelets* sísmicas podiam ser matematicamente modeladas. Em 1984, Crossman e Morlet estenderam este trabalho para mostrar como um sinal arbitrário pode ser decomposto em versões escalonadas e transladadas de uma única função de base, chamada *wavelet* mãe (ou, em inglês, *mother wavelet*). Propriedades das *wavelets* permitem, simultaneamente, análise no tempo e na frequência de um sinal, dado que as *wavelets* possuem energia concentrada no tempo, mas ainda preservam características periódicas das ondas. Em outras palavras, estas novas transformadas são mais localizadas

nestes dois domínios, diferentemente, por exemplo, das bases senoidais da análise de Fourier. Quanto à resolução tempo-frequencial, podem-se apresentar dois casos extremos: um impulso instantâneo, que possui exata localização no tempo, mas espectro espalhado; e uma onda senoidal, de frequência definida, mas energia espalhada no tempo. A transformada *wavelet* se apresenta como uma solução de compromisso entre estas características extremas: permite melhor localização na frequência do que o impulso, mas também possui melhor localização no tempo que a senóide, como ilustrado na Figura 3.15. Pode-se então dizer que existe uma relação recíproca entre a localização espacial e em frequência de uma função, de natureza similar ao princípio da incerteza de Heisenberg. De maneira simples, o preço que se paga pelo suporte compacto das *wavelets* é a perda de sua localização exata em frequência.

3.6.2 Modelagem

Wavelets são ondas localizadas, isto é, ao invés de oscilarem sempre sem qualquer decaimento (como uma senóide pura), caem para zero. Estas ondaletas, como são também chamadas, são obtidas da iteração ou cascadeamento de filtros (com reescalonamento). A ligação entre *wavelets* contínuas no tempo e banco de filtros discretos recai no limite de uma árvore logarítmica de filtros como ilustrado na Figura 3.16. Funções de escala - $\phi(t)$ (ou, em inglês, *Scaling functions*), e *wavelets* - $\omega(t)$, funções resultantes da reiteração dos canais passa-baixa e passa-alta, respectivamente, possuem propriedades desejáveis. Por exemplo, possuem a biortogonalidade (ou ortogonalidade) inerente ao banco de filtros que as originou.

Do ponto de vista matemático, *wavelets* são funções de base $\omega_{jk}(t)$ no tempo contínuo. Uma base é um conjunto de funções linearmente independentes que podem ser combinadas para se obter qualquer função admissível $f(t)$:

$$f(t) = \sum_{j,k} b_{jk} \omega_{jk}$$

Para vários propósitos é útil expandir uma dada função como uma combinação linear de algumas funções de base em particular. Um exemplo bastante conhecido é a análise de Fourier, em que as bases são $\{1, \sin(nx), \cos(nx) : n \in \mathcal{N}\}$. É fato que a análise de Fourier é uma ferramenta inestimável na análise de sinais periódicos como, por exemplo, vibrações. No entanto, para sinais não-periódicos ou para sinais com suporte compacto como, por exemplo, imagens, esse tipo de análise já não é a mais adequada. Isso se deve ao fato de as bases de Fourier não possuírem boa localização espacial, como mencionado. Assim, como as bases não decaem e é desejado que se anulem fora de um intervalo finito, a convergência se tornaria muito lenta.

Em casos como esse, uma abordagem melhor seria usar *wavelets*. A característica especial das bases *wavelet* é que todas as funções $\omega_{jk}(t)$ são construídas a partir de uma

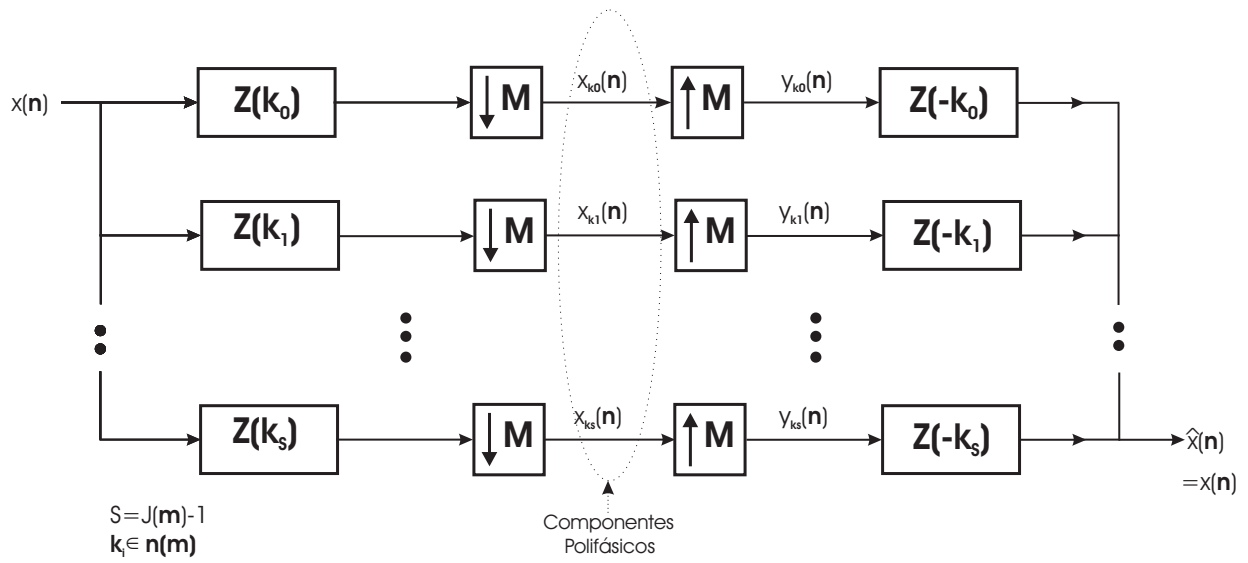


Figura 3.14: Decomposição de $x(n)$ em componentes polifásicos e sua recomposição.

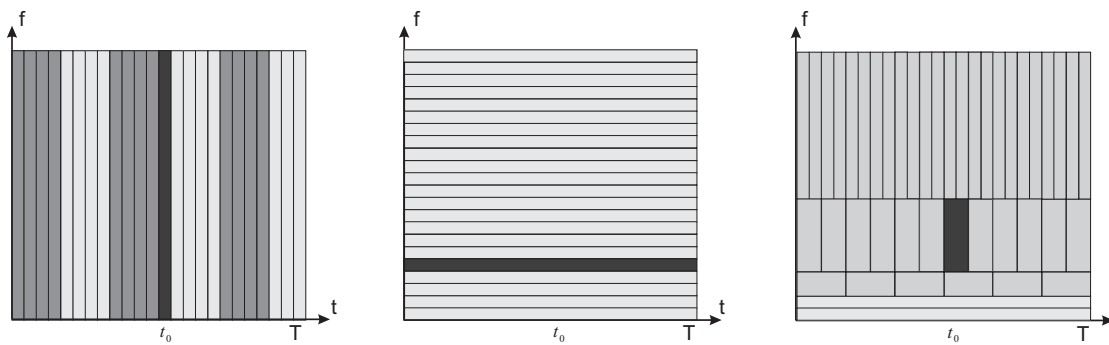


Figura 3.15: Representação tempo-frequencial para (a) impulso, (b) senóide pura e (c) *wavelet*.

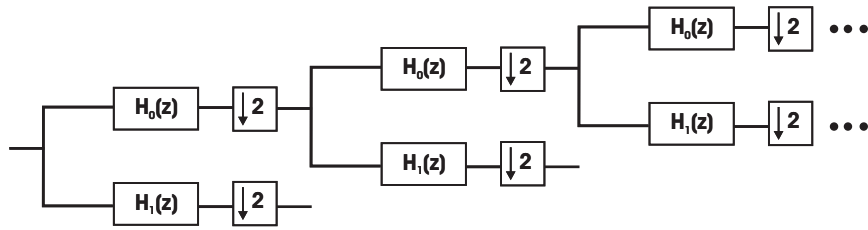


Figura 3.16: Estrutura em árvore logarítmica que origina as *wavelets*.

única *wavelet* mãe $\omega(t)$ por translação inteira (ou deslocamento), $\omega(t) \rightarrow \omega(t - 1)$, ou por dilatação diádica (ou reescalonamento), $\omega(t) \rightarrow \omega(2t)$. Essa *wavelet* é um pulso, isto é, normalmente começa no instante $t = 0$ e termina em um instante $t = N$. Os dois índices jk das funções de base $\omega_{jk}(t)$ representam, respectivamente, os deslocamentos e reescalonamentos da função mãe. Assim, as funções resultantes apenas de deslocamentos da função mãe serão representadas por $\omega_{0k}(t)$, começando no instante $t = k$ e terminando em $t = N + k$. Já as funções $\omega_{j0}(t)$, que resultam apenas de reescalonamentos da função mãe, se iniciam em $t = 0$ e terminam em $t = N/2^j$. O gráfico dessas funções reescaladas é, portanto, uma versão comprimida por um fator 2^j , enquanto o gráfico das *wavelets* deslocadas apenas é transladado de k :

$$\begin{aligned} \textit{wavelet} \text{ reescalada (gráfico comprimido): } & \omega_{j0}(t) = \omega(2^j t) \\ \textit{wavelet} \text{ deslocada (gráfico deslocado): } & \omega_{0k}(t) = \omega(t - k) \end{aligned}$$

Como consequência do repetido escalonamento, as *wavelets* decompõem um sinal em detalhes de várias escalas. Dada uma *wavelet* $\omega(t)$ e seu deslocamento puro $\omega(t - k)$, diz-se que ambas estão em escala unitária. Já as *wavelets* $\omega(2^j t)$ e $\omega(2^j t - k)$ estarão na escala 2^{-j} . É por essa propriedade que se diz que as *wavelets* produzem uma representação em multiresolução de uma imagem, preservando todos os detalhes importantes nela contidos. Uma consequência desse tipo de representação é que a saída do canal de mais baixa frequência da *wavelet* é uma versão compreensível do sinal de entrada, por isso também chamada de aproximação, diferentemente do que ocorre com a transformada de Fourier, por exemplo, em que a parte de mais baixa frequência normalmente é borrada. Essa idéia de multiresolução é uma característica importante das transformadas *wavelets*. Nessas transformadas, os sinais são decompostos em diferentes escalas de resolução, e não apenas em diferentes frequências. A tradicional análise "tempo-frequência" comumente aplicada a diferentes tipos de transformadas é então substituída por uma outra análise "tempo-escala" ao se tratar de *wavelets*. A multiresolução divide as frequências em oitavas, de ω a 2ω , ao invés de dividir em bandas uniformes, de ω a $\omega + \Delta\omega$.

Pode-se então considerar o caso mais geral, o da *wavelet* típica $\omega_{jk}(t)$, que é comprimida j vezes e escalonada k vezes. A fórmula que a descreve é:

$$\omega_{jk}(t) = \omega(2^j t - k)$$

A uma dada resolução j , pode-se dizer que as funções de escala $\phi(2^j t - k)$ formam as bases para a representação de um sinal ou de uma imagem. Já os detalhes obtidos nesse nível serão representados pelas *wavelets* $\omega(2^j t - k)$. Logo, o sinal suave mais os seus detalhes, as funções ϕ e ω , são combinados para se obter a representação em multiresolução do sinal para o próximo nível $j + 1$. Dessa maneira, médias podem ser associadas às funções de escala, enquanto os detalhes derivam das funções *wavelet*.

Já foi visto que as *wavelets* podem ser originadas de bancos de filtros organizados em árvore logarítmica como o da Figura 3.16. Nessa estrutura, nota-se que apenas a saída do canal passa-baixa foi sucessivamente decomposta. Logo, ao se optar por esse tipo de decomposição, pressupõe-se que a informação contida nas baixas frequências do sinal seria mais importante que a contida nas altas frequências. Para vários sinais isso não é necessariamente verdade. Por isso, pode-se optar por qualquer estrutura em árvore diádica, obtendo-se as chamadas bases *wavelet packet*. Nesse caso mais geral, em cada ponto da árvore tem-se a opção de enviar o sinal de saída do canal passa-baixa ou a do canal passa-alta, ou não enviar nada. A Figura 3.17 ilustra algumas dessas estruturas de decomposição. A primeira opção ilustrada é a já conhecida árvore logarítmica em que apenas o canal passa-baixa é decomposto novamente, correspondendo, portanto à estrutura mostrada na Figura 3.16. A segunda opção é o que se chama de árvore completa, análoga à *STFT* (do inglês, *Short Time Fourier Transform*). O método *Wavelet packets* permite que se origine toda essa família de bases. Cada partição pode ser associada a uma *quadtree*, dado que é caracterizada, em cada passo, por uma decisão de dividir ou não em dois cada canal. No caso de compressão de sinais, a decisão de dividir ou, eventualmente, fundir cada ramo deve ser motivada por critérios de taxa-distorção (ou, em inglês, *rate-distortion*).

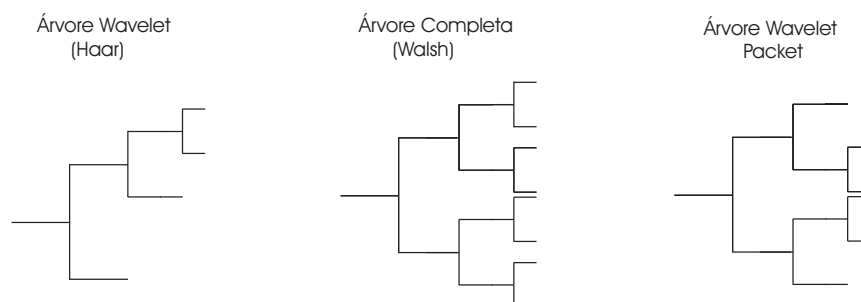


Figura 3.17: Cada estrutura *wavelet packet* origina um conjunto de bases, incluindo Walsh e Haar.

Capítulo 4

Transformadas direcionais implementadas via *lifting*

4.1 Histórico

Durante a última década, aumentou-se a sofisticação e a maturidade dos métodos de compressão de imagens baseados em *wavelets*. Com essa evolução, inicialmente pensou-se que a antiga transformada DCT teria sido definitivamente substituída pelas *wavelets*. O padrão internacional de codificação de imagens estáticas *JPEG* – 2000 [7], baseado em *wavelets*, não só apresentou melhor desempenho que o antigo padrão *JPEG* [5], baseado em DCT, como também passou a oferecer vantagens de escalabilidade em qualidade de reconstrução e em resolução espacial. Características essas desejáveis para várias aplicações em rede.

No entanto, a maioria das aplicações que utilizam *wavelets* bidimensionais (2-D DWT, do inglês, *Discrete Wavelet Transforms*), o fazem como um legado da transformada (2-D DCT), isto é, de forma separável. Como já discutido no Capítulo 2, esse método tradicional não é tão bom para representar elementos da imagem com direções arbitrárias, isto é, direções que não sejam horizontais ou verticais. Nesses casos, serão gerados coeficientes transformados de alta-frequência com grandes amplitudes, o que prejudica o desempenho dos algoritmos de compressão.

Note que, apesar da tendência em se considerar a *wavelet* uma verdadeira sucedânea da DCT, o mais novo padrão de codificação de vídeo H.264/AVC [26] é baseado em DCT. Vê-se que o uso dessa transformada se justifica, principalmente, pelo cálculo de baixa complexidade e também pelo fato de sua “pegada” (ou em inglês, “footprint”), ser menor que a das *wavelets*, ou seja, seu cômputo envolve uma vizinhança menor, possibilitando

eficiente transformação de blocos. Assim, não se pode dizer que a DCT foi superada pelas *wavelets*, mas sim que cada transformada possui características peculiares que devem ser compatibilizadas com a aplicação de interesse.

Como explorar a correlação direcional de uma imagem no domínio espacial ou frequencial tem sido um tópico de pesquisa há vários anos. Basicamente, os métodos direcionais de codificação baseados em *wavelets* podem ser divididos em duas grandes categorias:

- (a) Filtragem Direcional: Ikonomopoulos *et al.* [27] primeiro propuseram um banco de filtros 2-D que produz um componente passa-baixa da imagem e N componentes direcionais contendo altas frequências em uma dada direção. Li *et al.* [28] incorporaram a decomposição em sub-bandas ao trabalho de Ikonomopoulos, de tal forma que cada subbanda retangular contivesse detalhes em uma dada direção. Bamberger *et al.* [29] propuseram um banco de filtros criticamente decimado e regiões em formato de cunha (do inglês, *wedge*) para descrever a informação direcional, que recebeu bastante atenção por oferecer as características de decimação crítica e de reconstrução perfeita. Além desses, vários novos métodos foram propostos para preservar a informação direcional no domínio *wavelet*, como as *ridgelets* [30],[31], *curvelets* [22], *brushlets* [32], dentre outros. Esses filtros direcionais oferecem boas representações da informação direcional no domínio transformado, isto é, no domínio da frequência. Normalmente, esses métodos são utilizados em extração de características (ou, em inglês, *feature extraction*), no realce de imagens [33](ou, em inglês, *image enhancement*), e em extração de ruídos (ou, em inglês, *denoising*). No entanto, não são muito adequados para compressão, pois faltam codificadores de entropia eficientes para explorar a informação direcional em cada região da *wavelet*.
- (b) Predição Direcional: A questão central ao se aplicar predição direcional em decomposições *wavelet* é o conflito entre transformada global e direcionalidades localizadas. Imagens naturais normalmente apresentam fortes direcionalidades localizadas. Pode-se então partir uma imagem em várias regiões pequenas, aproveitando-se a correlação entre as direções. No entanto, note que isso causará severos efeitos de borda, prejudicando a eficiência da codificação por *wavelets*. Taubman *et al.* [34] propuseram uma técnica para reamostrar a imagem antes de realizar uma decomposição convencional em sub-bandas, rotacionando as bordas da imagem para alinhá-las horizontal ou verticalmente. Outros métodos propõem separar a imagem em partes e utilizar cada parte para prever a outra. Normalmente, esses métodos levam a transformações não-separáveis. A transformada implementada nesse trabalho insere técnicas de predição em direções arbitrárias no contexto de transformadas *wavelet* (2D) baseadas em *lifting*. A estrutura *lifting*, baseada em estágios canônicos obtidos através da decomposição de Smith [10], é uma maneira eficiente de se implementar uma transformada *wavelet* em que a resposta impulsional de cada filtro FIR pode ser fatorada em alguns estágios [35]. Recentemente, abordagens em que as direções

de filtragem são selecionadas de maneira adaptativa e explicitamente por *lifting* têm sido propostas [36], [37], [38], [39], [40]. Ding *et al.* [40] propôs uma transformada direcional adaptativa implementada via *lifting* (ADL, do inglês *Adaptive Directional Lifting*), incorporando a informação direcional aos estágios de *lifting*, abordagem similar à utilizada nesse trabalho. O método ADL permite que se insira a informação direcional em qualquer filtro de *wavelet* unidimensional, como os populares *5/3-taps* ou o *9/7-taps* [7], para se realizar decomposições direcionais bidimensionais. Algumas das abordagens se utilizam apenas de deslocamentos inteiros para rotacionarem os filtros convencionais, enquanto outras se utilizam de métodos *sub-pixel*, que requerem interpolações igualmente direcionais.

Terminado o histórico recente sobre o desenvolvimento de transformadas *wavelets* direcionais, será dada ênfase no entendimento do esquema *lifting*, pois é nessa abordagem em que o trabalho desenvolvido se baseia.

4.2 *Lifting*

Lifting é um procedimento utilizado para projetar DWT's, que garante reconstrução perfeita [3], [4]. Foi mostrado [3] que qualquer DWT biortogonal de duas bandas pode ser fatorada em pares de etapas de *lifting*. Nessa pesquisa, no entanto, apenas serão consideradas transformadas *wavelet* que podem ser implementadas com apenas um par de etapas de *lifting*, isto é, uma etapa de predição (ou, em inglês, *Prediction*), e uma de atualização (ou, em inglês, *Update*). Exemplos de *wavelets* que se inserem nessa categoria são a *wavelet* de Haar, a *wavelet* 5/3 e a família de *wavelets* interpolantes [41]. Note que a difundida *wavelet* cdf 9/7 não foi citada, visto que sua fatoração envolve dois pares de etapas [3]. Esse detalhe será evidenciado mais à frente em um exemplo de decomposição.

Esse método de implementação de transformadas *wavelet* permite adaptar localmente a direção dos filtros, aproveitando melhor as direcionalidades contidas nas imagens. A Figura 4.1 ilustra o resultado da filtragem da imagem Bárbara por um filtro de Gabor. Filtros de Gabor possuem, simultaneamente, seletividade na frequência e na orientação, além de apresentar resolução ótima conjunta espaço-temporal [42], características desejáveis para ressaltar as direcionalidades existentes em uma imagem. A Figura 4.2, ilustra as direcionalidades espaciais encontradas nos blocos de uma imagem exemplificativa, identificadas após procedimentos de pré-processamento e filtragem de Gabor. Deve-se notar que há um número considerável de blocos com direcionalidades dominantes diferentes. Logo, vê-se que realmente existe potencial em se tentar representar melhor essas direcionalidades.

Imagem filtrada com filtro de Gabor



Figura 4.1: Resultado da filtragem de um filtro adaptativo de Gabor da Imagem Bárbara.

4.2.1 Modelagem

A idéia básica das transformadas *wavelet* é a de explorar a correlação existente na maioria dos sinais reais, representando-os de maneira esparsa. A estrutura de correlação é tipicamente localizada em espaço e em frequência, isto é, amostras espacialmente vizinhas e amostras de frequências vizinhas são mais correlacionadas do que se estivessem distantes. O projeto tradicional de *wavelets* usa a transformada de Fourier para construir a localização espaço-frequencial. No entanto, como será mostrado no um exemplo unidimensional abaixo, isso também pode ser feito no domínio espacial.

Considere um sinal $\{x(n)\}$ que, de forma análoga ao que foi feito com filtros no Capítulo 3, pode ser separado em dois conjuntos disjuntos, que são chamados componentes polifásicos: o conjunto das amostras com índices pares $\mathcal{X}_p = \{x(2n)\}$, e o conjunto de amostras com índices ímpares $\mathcal{X}_i = \{x(2n+1)\}$. Essa etapa é conhecida como passo de separação (ou, em inglês, *Split*). Tipicamente, esses dois conjuntos são bastante correlacionados, de forma que possa construir um bom preditor P para um dos conjuntos, o ímpar, por exemplo, a partir do outro, o par. Dificilmente será obtido um preditor perfeito, logo, torna-se necessário armazenar o resíduo ou o detalhe $d(n)$:

$$d(n) = x(2n+1) - P_n(\mathcal{X}_p). \quad (4.1)$$



Figura 4.2: Imagem Bárbara binarizada com vetores indicando as direções predominantes nos blocos, obtidas por meio de filtros de Gabor.

Assim, dado o detalhe $d(n)$ e as amostras ímpares, essas podem ser perfeitamente recuperadas por:

$$x(2n + 1) = P_n(\mathcal{X}_p) + d(n). \quad (4.2)$$

Quanto melhor for o preditor P , mais esparsos serão o conjunto $\mathcal{D} = \{d(n)\}$. Em outras palavras, espera-se que a entropia de \mathcal{D} seja menor que a de \mathcal{X}_i . Um preditor bem simples para uma amostra ímpar x_{2k+1} seria a média de suas amostras pares vizinhas. O coeficiente de detalhe seria, então:

$$d(n) = x(2n + 1) - (x(2n) + x(2n + 2))/2. \quad (4.3)$$

Pode-se perceber que, nesse caso, se o sinal for localmente linear, o coeficiente de detalhe será nulo. A operação de calcular a predição e de guardar o resíduo é o que se chama estágio ou passo de predição de *lifting*. A idéia de se guardar o coeficiente de detalhe $d(n)$ ao invés

da própria amostra ímpar $x(2n + 1)$ é bastante conhecida, e é a base para os chamados métodos DPCM (do inglês, *Differential Pulse Code Modulation*) [43],[44]. O passo de predição já consegue aproveitar parte da correlação espacial, porém, para se constituir uma *wavelet*, ainda se deve obter alguma separação em frequência. No momento, tem-se uma transformação de $(\mathcal{X}_p, \mathcal{X}_i)$ em $(\mathcal{X}_p, \mathcal{D})$. Logo, a separação em frequência ainda é pobre, visto que \mathcal{X}_i foi obtido por simples subamostragem, o que causa o já citado efeito de *aliasing*. Para corrigir isso, foi proposto um segundo passo de *lifting*, o chamado passo de atualização, que usa um operador U aplicado aos detalhes para substituir as amostras pares por valores suaves $s(n)$:

$$s(n) = x(2n) + U_n(\mathcal{D}). \quad (4.4)$$

Novamente, esse é um passo inversível: dados (\mathbf{s}, \mathbf{d}) , x_e pode ser recuperado por:

$$x(2n) = s(n) - U_n(\mathcal{D}), \quad (4.5)$$

enquanto $x(2n + 1)$ pode ser recuperado como em 4.2. Isso ilustra uma das características mais importantes do *lifting*: não importa como sejam escolhidos P e U , as operações são sempre inversíveis, o que implica bancos de filtros com reconstrução perfeita. O diagrama de blocos da Figura 4.3 ilustra os três passos de *lifting*: *Split*, *Prediction-P* e *Update-U*.

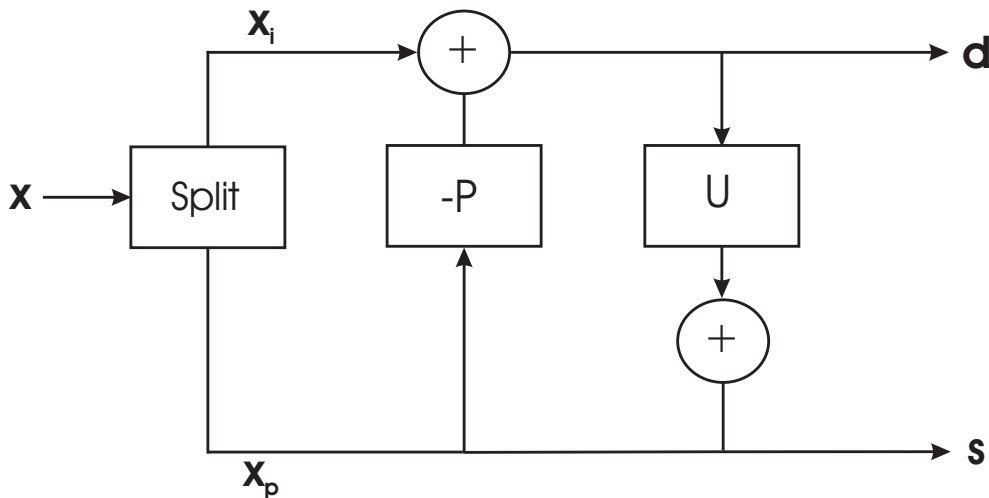


Figura 4.3: Representação em diagrama de blocos dos três passos de *lifting*: *Split*, *Prediction-P* e *Update-U*.

Uma questão pertinente é quais tipos de *wavelets* poderiam ser decompostas em passos de *lifting*. De fato, qualquer *wavelet* ou banco de filtros FIR pode ser decomposto dessa maneira. Isso pode ser verificado escrevendo-se a transformada em sua forma polifásica, estudada no Capítulo 3. Assim, considerações sobre reconstrução perfeita ou *lifting* poderão ser feitas utilizando-se matrizes contendo polinômios (simplesmente), ou polinômios de

Laurent. O chamado passo de *lifting* se torna então uma matriz elementar, isto é, uma matriz triangular (superior ou inferior), com os elementos da diagonal unitários. É um resultado bastante conhecido em álgebra de matrizes que qualquer matriz com elementos polinômios e determinante unitário pode ser fatorada em matrizes elementares. A prova recai sobre o algoritmo de Euclides, desenvolvido há 2000 anos. Na literatura de bancos de filtros, vários trabalhos se utilizam desse algoritmo para construir transformadas em sub-bandas a partir de matrizes elementares. Por exemplo, Vetterli e Herley [45] usam o algoritmo de Euclides, e sua conexão com equações Diofantinas, para achar todos os filtros passa-alta que, dado um filtro passa-baixa, resultam em transformadas *wavelets* com filtros finitos.

Por fim, será feito um exemplo de fatoração da matriz polifásica em passos de *lifting*, usando-se a popular transformada *wavelet* cdf 9/7. Sabe-se que 9 e 7 são, respectivamente, a quantidade de *taps* ou de coeficientes dos filtros passa-baixa (h) e passa-alta (g) de análise. Assim, seja o filtro simétrico não-causal:

$$h(z) = h_4(z^4 + z^{-4}) + h_3(z^3 + z^{-3}) + h_2(z^2 + z^{-2}) + h_1(z + z^{-1}) + h_0. \quad (4.6)$$

Como mostrado no Capítulo 3, esse filtro pode ser decomposto em seus componenes polifásicos:

$$h(z) = h_p(z^2) + z^{-1}h_i(z^2), \quad (4.7)$$

em que $h_p(z) = h_4(z^2 + z^{-2}) + h_2(z + z^{-1}) + h_0$ e $h_i(z) = h_3(z^2 + z^{-1}) + h_1(z + 1)$.

De forma similar, seja o seguinte filtro passa-alta:

$$g(z) = g_3(z^2 + z^{-4}) + g_2(z + z^{-3}) + g_1(1 + z^{-2}) + g_0z^{-1}, \quad (4.8)$$

segue-se sua decomposição polifásica:

$$g(z) = g_p(z^2) + z^{-1}g_i(z^2), \quad (4.9)$$

em que $g_p(z) = g_3(z + z^{-2}) + g_1(1 + z^{-1})$ e $g_i(z) = g_2(z + z^{-1}) + g_0$.

Prossegue-se então à montagem da matriz polifásica de análise $\mathbf{P}(z)$. De acordo com o exposto no Capítulo 3, para esse banco de filtros, com $M = 2$, tem-se:

$$\mathbf{P}(z) = \begin{bmatrix} h_p(z) & h_i(z) \\ g_p(z) & g_i(z) \end{bmatrix} = \begin{bmatrix} h_4(z^2 + z^{-2}) + h_2(z + z^{-1}) + h_0 & h_3(z^2 + z^{-1}) + h_1(z + 1) \\ g_3(z + z^{-2}) + g_1(1 + z^{-1}) & g_2(z + z^{-1}) + g_0 \end{bmatrix}. \quad (4.10)$$

Aplicando-se o algoritmo de Euclides [4], foram obtidos os seguintes restos das divisões dos polinômios:

$$\begin{aligned} r_0 &= h_0 - 2h_4h_1/h_3 \\ r_1 &= h_2 - h_4 - h_4h_1/h_3 \\ s_0 &= h_1 - h_3 - h_3r_0/r_1 \\ t_0 &= r_0 - 2r_1. \end{aligned}$$

Definindo-se os parâmetros:

$$\begin{aligned} \alpha &= h_4/h_3 \approx -1,586134342 \\ \beta &= h_3/r_1 \approx -0,05298011854 \\ \gamma &= r_1/s_0 \approx 0,8829110762 \\ \delta &= s_0/t_0 \approx 0,4435068522 \\ \zeta &= t_0 = r_0 - 2r_1 \approx 1,149604398 \end{aligned}$$

Chega-se à seguinte fatoração da matriz polifásica de análise:

$$\mathbf{P}(z) = \begin{bmatrix} \zeta & 0 \\ 0 & 1/\zeta \end{bmatrix} \begin{bmatrix} 1 & \delta(1+z) \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ \gamma(1+z^{-1}) & 1 \end{bmatrix} \begin{bmatrix} 1 & \beta(1+z) \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ \alpha(1+z^{-1}) & 1 \end{bmatrix}. \quad (4.13)$$

A fatoração leva à seguinte implementação espacial, em que os índices sobrescritos representam o número da iteração:

$$\begin{aligned} s(n)^{(0)} &= x(2n) \\ d(n)^{(0)} &= x2n + 1 \\ d(n)^{(1)} &= d(n)^{(0)} + \alpha(s(n)^{(0)} + s(n+1)^{(0)}) \\ s(n)^{(1)} &= s(n)^{(0)} + \beta(d(n)^{(1)} + d(n-1)^{(1)}) \\ d(n)^{(2)} &= d(n)^{(1)} + \gamma(s(n)^{(1)} + s(n+1)^{(1)}) \\ s(n)^{(2)} &= s(n)^{(1)} + \delta(d(n)^{(2)} + d(n-1)^{(2)}) \\ s(n) &= \zeta s(n)^{(2)} \\ d(n) &= d(n)^{(2)}/\zeta \end{aligned} \quad (4.14)$$

A partir do exposto, podem ser identificados dois passos de predição, $P1$ e $P2$, e dois passos de atualização, $U1$ e $U2$, tais que:

$$\begin{aligned} P1 &= -\alpha(s(n)^{(0)} + s(n+1)^{(0)}) & P2 &= -\gamma(s(n)^{(1)} + s(n+1)^{(1)}) \\ U1 &= \beta(d(n)^{(1)} + d(n-1)^{(1)}) & U2 &= \delta(d(n)^{(2)} + d(n-1)^{(2)}) \end{aligned} \quad (4.15)$$

Nesse exemplo, puderam ser lembrados os conceitos de decomposição polifásica, montando-se a matriz polifásica de análise de um banco de filtros unidimensional com 2

bandas a partir desses componentes. Em seguida, procedeu-se à fatoração dessa matriz em matrizes elementares, que compõem os chamados passos de *lifting*. Por fim, evidenciou-se a implementação espacial dessa transformada. Na próxima seção, será feita uma extensão desses conceitos para o caso bidimensional, dado o foco central do trabalho ser a aplicação desse método de implementação de transformadas *wavelet* em imagens.

4.3 *Lifting* bidimensional convencional

Relembrando o procedimento descrito no Capítulo 2, uma DWT bidimensional implementada de forma separável, em geral, consiste de dois estágios. No primeiro estágio, uma DWT unidimensional 1D-DWT é aplicada à imagem seguida de subamostragem vertical, por exemplo, para se obter a sub-banda passa-baixa, L , e a sub-banda passa-alta H . No segundo estágio, novamente será aplicada uma (1D-DWT) às sub-bandas L e H seguida de subamostragem horizontal, resultando, respectivamente, nas sub-bandas LL e LH , e nas sub-bandas HL e HH .

A modelagem matemática desse procedimento será iniciada com a representação da imagem original digitalizada, considerando-se que foi realizada uma amostragem do tipo retangular. Assim, seja $\mathbf{n} = (n_x, n_y)^T$, o vetor que representa a localização de uma amostra da grade bidimensional $\mathcal{G} = \{(n_x, n_y)^T \in \mathcal{Z}^2\}$, cuja intensidade é $x(\mathbf{n})$, o conjunto de amostras pode ser representado por $\mathcal{X} = \{x(\mathbf{n}) | \mathbf{n} \in \mathcal{G}\}$. Pode-se, então, decompor a grade \mathcal{G} em quatro sub-grades $\mathcal{G}_{pq} = \{(n_x, n_y)^T \in \mathcal{G} | n_y \bmod 2 = p, n_x \bmod 2 = q\}$, com $p = 0, 1$ e $q = 0, 1$. Separação essa que é conveniente, pois agrupa amostras localizadas em linhas pares e colunas pares na sub-grade (\mathcal{G}_{00}); linhas pares e colunas ímpares na sub-grade (\mathcal{G}_{01}); linhas ímpares e colunas pares na sub-grade (\mathcal{G}_{10}); e linhas ímpares e colunas ímpares na sub-grade (\mathcal{G}_{11}).

Para se aplicar a (2-D DWT) via *lifting*, inicia-se aplicando a transformada entre as linhas pares e ímpares, isto é, entre $\mathcal{X}_0 = \{x(\mathbf{n}_0) | \mathbf{n}_0 \in \mathcal{G}_0 = \mathcal{G}_{00} \cup \mathcal{G}_{01}\}$ e $\mathcal{X}_1 = \{x(\mathbf{n}_1) | \mathbf{n}_1 \in \mathcal{G}_1 = \mathcal{G}_{10} \cup \mathcal{G}_{11}\}$. Designando-se a sub-banda passa-baixa resultante por $\mathcal{W}_0 = \{w_0(\mathbf{n}_0) | \mathbf{n}_0 \in \mathcal{G}_0\}$, e a sub-banda passa-alta por $\mathcal{W}_1 = \{w_1(\mathbf{n}_1) | \mathbf{n}_1 \in \mathcal{G}_1\}$, esse primeiro estágio da transformada em \mathcal{X} pode ser genericamente expresso por:

$$w_1(\mathbf{n}_1) = g_H (x(\mathbf{n}_1) - P_{\mathbf{n}_1}(\mathcal{X}_0)), \forall \mathbf{n}_1 \in \mathcal{G}_1 \quad (4.16)$$

$$w_0(\mathbf{n}_0) = g_L (x(\mathbf{n}_0) + g_H^{-1} U_{\mathbf{n}_0}(\mathcal{X}_1)), \forall \mathbf{n}_0 \in \mathcal{G}_0 \quad (4.17)$$

em que as funções de Predição e de Atualização, $P_{\mathbf{n}_1}(\cdot)$ e $U_{\mathbf{n}_0}(\cdot)$ são, respectivamente, funções das amostras de entrada com saída escalar, e g_H e g_L são fatores de escala.

De maneira análoga, o segundo estágio aplicará a transformada entre as colunas pares e ímpares, decompondo finalmente \mathcal{W}_0 em \mathcal{W}_{00} (definida em \mathcal{G}_{00}), e em \mathcal{W}_{01} (definida em \mathcal{G}_{01}); e \mathcal{W}_1 em \mathcal{W}_{10} (definida em \mathcal{G}_{10}), e em \mathcal{W}_{11} (definida em \mathcal{G}_{11}). Portanto, \mathcal{W}_{00} , \mathcal{W}_{01} , \mathcal{W}_{10} e \mathcal{W}_{11} equivalem, respectivamente, às sub-bandas LL , LH , HL e HH . Todo o processo descrito para decompor o conjunto das amostras originais de uma imagem \mathcal{X} em quatro sub-bandas, é conhecido como um nível completo de (2-D DWT). Múltiplos níveis podem ser obtidos pela decomposição sucessiva da sub-banda LL resultante.

Note que, nessa seção, considerou-se a filtragem bidimensional como resultante de duas filtragens unidimensionais sucessivas, isto é, um procedimento separável. Dessa maneira, quando as linhas pares foram separadas das ímpares, por exemplo, ao se obter um preditor para uma dada amostra $x(\mathbf{n}) = x(n_{k1}, n_{k2})$, a função de predição correspondente apenas pôde acessar amostras em linhas diferentes, localizadas, porém, sob a mesma coluna n_{k2} . O mesmo se aplica à função de atualização. Situação análoga é verificada quando se realiza a transformada entre as colunas. No entanto, dessa vez as funções de predição e de atualização terão seus domínios restritos a amostras localizadas na mesma linha da amostra a ser predita ou atualizada. A Figura 4.4 mostra possíveis amostras a serem acessadas nessas duas etapas. Na próxima seção será mostrado um método direcional em que essas restrições não serão observadas. Isso permitirá melhor aproveitamento das direcionalidades da imagem, porém, tornará o método não-separável.

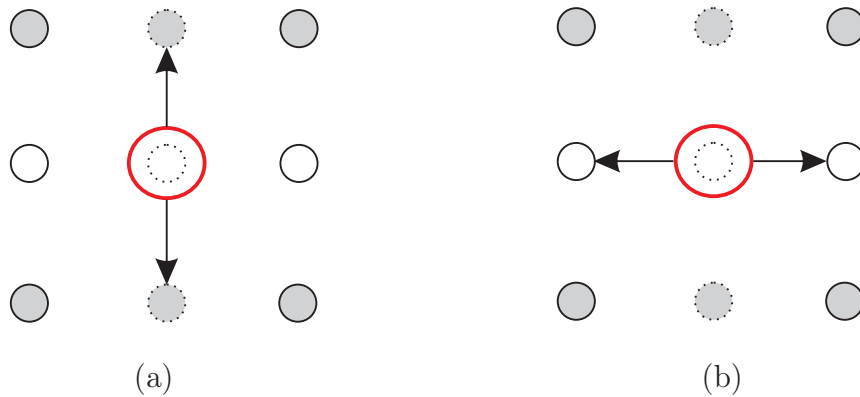


Figura 4.4: Representação das direções de predição e de atualização de uma (2-D DWT) convencional separável. As circunferências sólidas não preenchidas representam \mathcal{G}_{00} , as tracejadas não preenchidas \mathcal{G}_{01} , as sólidas preenchidas \mathcal{G}_{10} e as tracejadas preenchidas \mathcal{G}_{11} . Em destaque encontram-se as amostras a serem preditas ou atualizadas. (a) Transformação entre linhas: Apenas amostras localizadas sob a mesma coluna poderão compor as funções de predição e de atualização. (b) Transformação entre colunas: Apenas amostras localizadas sob a mesma linha poderão compor as funções de predição e de atualização.

4.4 *Lifting* bidimensional direcional

Em uma 2D-DWT convencional, as funções de Predição em 4.16 e de Atualização em 4.17 podem ser expressas por:

$$P_{\mathbf{n}_1}(\mathcal{X}_0) = \sum_{k=-K_P}^{K_P-1} c_{P,k} \cdot x(n_{1,x}, n_{1,y} - (2k + 1)) \quad (4.18)$$

$$U_{\mathbf{n}_0}(\mathcal{W}_1) = \sum_{k=-K_U}^{K_U-1} c_{U,k} \cdot w_1(n_{0,x}, n_{0,y} - (2k + 1)) \quad (4.19)$$

em que K_P , c_P, k , K_P e c_U, k são determinados pelo núcleo ou *kernel* da *wavelet* adotada. Observe que, nesse caso, as funções foram expressas de forma a atender a restrição mencionada: no primeiro estágio da transformação, realizada ao longo das linhas, por exemplo, apenas amostras localizadas em uma mesma coluna são envolvidas. Para que se possa atingir a finalidade de compressão e considerando-se cada amostra da sub-banda passa-alta, $w_1(\mathbf{n}_1)$, é desejado obter uma função de predição $P_{\mathbf{n}_1}(\mathcal{X}_0)$ em 4.16 que seja capaz de prever $x(\mathbf{I}_1)$ a partir das amostras em \mathcal{X}_0 com um resíduo de menor amplitude possível. No método adaptativo proposto, definem-se filtros direcionais de predição em função da direção $\mathbf{d} = (d_x, d_y)$, que deve ser selecionada adaptativamente por:

$$P_{\mathbf{n}_1, \mathbf{d}}(\mathcal{S}_0) = \sum_{k=-K_P}^{K_P-1} c_{P,k} \cdot x(\mathbf{n}_1 - (2k + 1) \mathbf{d}) \quad (4.20)$$

em que \mathbf{d} é definida tal que:

$$\mathbf{n}_1 - (2k + 1) \mathbf{d} \in \mathcal{G}_0, \forall \mathbf{n}_1 \in \mathcal{G}_1, k = -K_P, \dots, K_P - 1. \quad (4.21)$$

O filtro direcional de predição corresponde à realização do passo de predição ao longo da direção \mathbf{d} , ou seja, as amostras de entrada dessa função serão selecionadas ao longo dessa direção escolhida. Desta maneira, identificando-se uma região mais homogênea na imagem ao longo de uma direção predominante, deve-se adotar essa direção para compor o filtro de predição, implicando em resíduos menores, já que os *pixels* pertencentes a essa região são mais parecidos entre si do que com os vizinhos localizados abaixo e acima, ou aos lados. De 4.21, vê-se que $\mathbf{d} \in \mathcal{Z}^2$ e d_y é sempre ímpar. Estabeleceremos uma restrição a \mathbf{d} , de tal forma que o segmento de reta de $(0, 0)$ até (d_x, d_y) não intercepte qualquer outra amostra da grade, isto é, que d_x e d_y sejam primos entre si. Tipicamente, utilizaremos nove direções a serem testadas:

$$\begin{aligned}
\mathbf{d}_{-4} &= (-3, 1)^T, & \mathbf{d}_{-3} &= (-2, 1)^T, & \mathbf{d}_{-2} &= (-1, 1)^T, \\
\mathbf{d}_{-1} &= (-1, 3)^T, & \mathbf{d}_0 &= (0, 1)^T, & \mathbf{d}_1 &= (1, 3)^T, \\
\mathbf{d}_2 &= (1, 1)^T, & \mathbf{d}_3 &= (2, 1)^T & \mathbf{d}_4 &= (3, 1)^T.
\end{aligned} \tag{4.22}$$

A disposição dessas direções na grade está ilustrada na Figura 4.5.

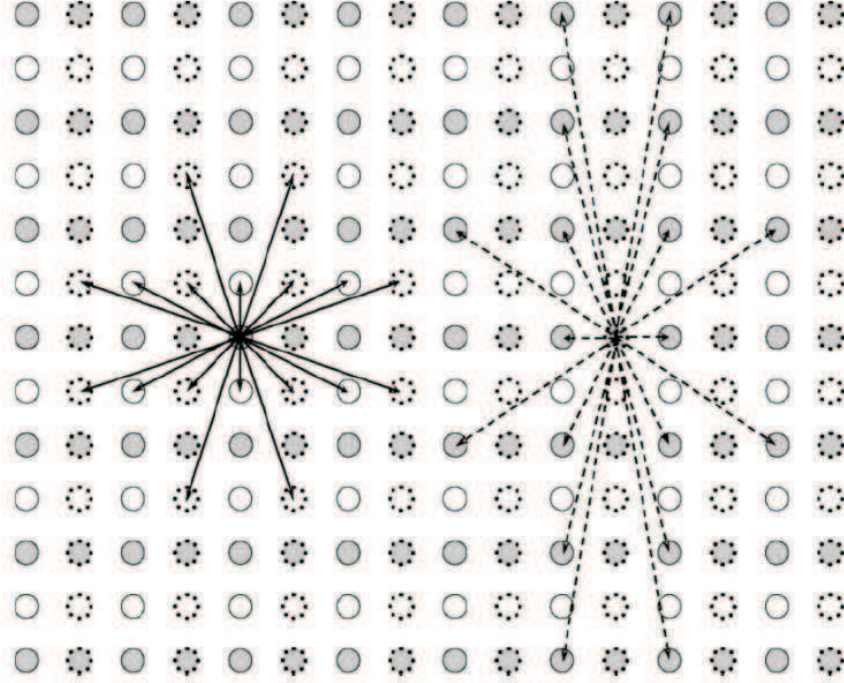


Figura 4.5: Conjunto de direções propostas para a (2-D DWT) direcional adaptativa. as flechas sólidas representam os vetores \mathbf{d} e $-\mathbf{d}$, enquanto os tracejados representam $\tilde{\mathbf{d}}$ e $-\tilde{\mathbf{d}}$. As circunferências sólidas não preenchidas representam \mathcal{G}_{00} , as tracejadas não preenchidas \mathcal{G}_{01} , as sólidas preenchidas \mathcal{G}_{10} , e as tracejadas preenchidas \mathcal{G}_{11} .

Designa-se $\mathbf{d}_{\mathbf{n}_1}^*$ a direção escolhida na localização \mathbf{n}_1 para $P_{\mathbf{n}_1}(\mathcal{S}_0)$. Assim, após completar o passo de predição para todas as amostras, segue o passo de atualização, com a correspondente função definida por:

$$U_{\mathbf{n}_0}(\mathcal{W}_1) = \sum_{k=-K_U}^{K_U-1} c_{U,k} \sum_{\{\mathbf{n}_1 | \mathbf{n}_1 - (2k+1)\mathbf{d}_{\mathbf{n}_1}^* = \mathbf{n}_0\}} w_1(\mathbf{n}_1). \tag{4.23}$$

Em outras palavras, enquanto uma amostra da imagem é predita por $c_{P,k}x(\mathbf{n}_0)$, $x[\mathbf{n}_0]$ é atualizada por $c_{U,k}w_1(\mathbf{n}_1)$. Note que, se a direção \mathbf{d}_0 for sempre selecionada, as equações 4.20 e 4.23 equivalem, respectivamente, a 4.18 e a 4.19, e o *lifting* direcional se torna igual ao convencional. Dito de outra maneira, o *lifting* direcional é uma generalização do convencional. O segundo estágio de DWT, ou a transformação ao longo das colunas, deve ser aplicado a \mathcal{W}_0 e a \mathcal{W}_1 , e a transformação deve ser feita ao longo de novas direções $\tilde{\mathbf{d}}$,

que decorrem do conjunto de direções \mathbf{d} do primeiro estágio, como mostrado na Figura 4.5. O novo conjunto de direções é então especificado por: $\check{\mathbf{d}} = (\check{d}_x, \check{d}_y)^T = (d_y, 2d_x)^T$, isto é, $\check{\mathbf{d}}_{-4} = (1, -6)^T$, $\check{\mathbf{d}}_{-3} = (1, -4)^T$, $\check{\mathbf{d}}_{-2} = (1, -2)^T$, $\check{\mathbf{d}}_{-1} = (3, -2)^T$, $\check{\mathbf{d}}_0 = (1, 0)^T$, $\check{\mathbf{d}}_1 = (3, 2)^T$, $\check{\mathbf{d}}_2 = (1, 2)^T$, $\check{\mathbf{d}}_3 = (1, 4)^T$ e $\check{\mathbf{d}}_4 = (1, 6)^T$. Conseqüentemente, \check{d}_y é par e \check{d}_x é ímpar, e assim:

$$\begin{aligned} \mathbf{n}_{01} - (2k + 1)\check{\mathbf{d}} &\in \mathcal{G}_{00}, & \mathbf{n}_{11} - (2k + 1)\check{\mathbf{d}} &\in \mathcal{G}_{10} \\ \forall \mathbf{n}_{01} &\in \mathcal{G}_{01}, & \mathbf{n}_{11} &\in \mathcal{G}_{11} \\ & & k &= -K_P, \dots, K_P - 1. \end{aligned} \quad (4.24)$$

Em princípio, as direções de transformação de $w_0(\mathbf{n})$ e $w_1(\mathbf{n})$ podem ser diferentes, e as funções de predição e atualização são definidas de forma análoga a 4.20 e 4.23. Dado que $\check{\mathbf{d}} = (\check{d}_x, \check{d}_y)^T = (d_y, 2d_x)^T$, a implementação do *lifting* direcional em \mathcal{W}_0 e \mathcal{W}_1 , ao longo de $\check{\mathbf{d}}_i$, pode ser entendida como a mesma implementação do *lifting* direcional em \mathcal{X} , ao longo de \mathbf{d}_i , se primeiro \mathcal{W}_0 e \mathcal{W}_1 forem verticalmente subamostrados por um fator de 2 e transpostos.

Na definição do núcleo (ou, em inglês, *kernel*), de *wavelet* a ser utilizado, adotou-se a *wavelet* interpolante (6,6), como sugerido em [37]. Assim, obtêm-se: $K_P = 3$, com $c_{p0} = c_{p-1} = 150/256$, $c_{p1} = c_{p-2} = -25/256$, $c_{p2} = c_{p-3} = 3/256$, e $K_U = 3$, com $c_{u0} = c_{u-1} = 150/512$, $c_{u1} = c_{u-2} = -25/512$, $c_{u2} = c_{u-3} = 3/512$. Correspondentemente, fez-se $g_L = \sqrt{1.706}$ e $g_H = \sqrt{0.626}$ a partir dos filtros de síntese, de tal forma que, se o ruído de quantização dos coeficientes transformados for descorrelacionado, a energia do ruído é preservada na imagem reconstruída.

Existem outras abordagens na literatura, também capazes de prover adaptações direcionais via *lifting*, que adotam um conjunto de direções sub-*pixel*, isto é, o deslocamento na grade não é inteiro como no conjunto de direções \mathbf{d} apresentado [46], [40]. Nessas abordagens, normalmente, o deslocamento na grade é associado à tangente de uma inclinação espacial, como sugerido na Figura 4.6. Note que a quantidade de possíveis ângulos a serem considerados depende da quantidade de amostras a serem acrescentadas, isto é, para uma precisão de meio *pixel* se terá uma quantidade menor de ângulos se comparada à precisão de quarto de *pixel*. Em princípio, poder-se-ia imaginar que as abordagens sub-*pixel* seriam capazes de prover uma representação mais acurada. No entanto, é preciso notar que, nesse caso, torna-se necessário interpolar os *pixels* originais da imagem, o que pode descaracterizar a direcionalidade original. Existem alguns trabalhos propondo resolver esse problema por meio de uma interpolação igualmente direcional, o que também implica um descasamento (ou em inglês, *mismatch*) entre os passos de predição e de atualização [46]. Além disso, dado que a abordagem adotada não realiza interpolações, essa é, computacionalmente, menos complexa. Logo, não é tão evidente dizer qual abordagem é a mais precisa.

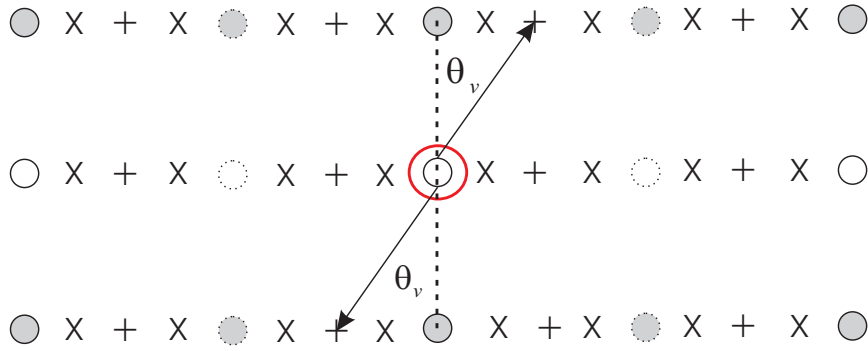


Figura 4.6: Representação de métodos de implementação de 2-D DWT's direcionais adaptativas em que a direção é escolhida a partir da tangente de uma inclinação espacial θ_v , envolvendo, portanto, localizações sub-*pixel*. As posições inteiras são marcadas por “O”, as posições *half-pixel* são marcadas por “+”, enquanto as posições *quarter-pixel* são marcadas por “x”.

4.4.1 Escolha adaptativa das direções

Para fins de compressão de imagem, as direções de filtragem na 2-D DWT devem ser escolhidas de forma a minimizar a distorção da imagem reconstruída para uma dada taxa, algoritmo bastante conhecido como otimização taxa-distorção (ou, em inglês, *Rate-Distortion Optimization*). Observe que, no caso da transformada *wavelet* adaptativa, a taxa deve indicar também a direção escolhida, além de representar os próprios coeficientes transformados. No entanto, deve-se lembrar que, no momento em que se deseja escolher a melhor direção em um bloco, não se dispõe da imagem inteira decomposta, inviabilizando que sejam incluídos na taxa os *bits* gastos na representação dos coeficientes. Assim, achar a direção ótima é similar ao que se faz em codificação de vídeo, estimando-se a direção de movimento dos blocos de um quadro (ou, em inglês, *frame*) para outro, com restrição de taxa [47],[48].

Considerando-se o primeiro estágio de transformada em \mathbf{s} , a grade original $\mathbf{\Pi}$ pode ser dividida em N_b blocos não sobrepostos, cada um designado por \mathbf{B}_b , $b = 0 \dots N_b - 1$. Para cada bloco, a direção ótima $\bar{\mathbf{d}}_b$ pode ser escolhida minimizando-se uma função de custo Lagrangeana:

$$\bar{\mathbf{d}}_b = \arg \min_{\mathbf{d}} \left\{ \sum_{\mathbf{n}_1 \in \mathcal{G}_1 \cap \mathbf{B}_b} D(g_H(x[\mathbf{n}_1] - P_{\mathbf{n}_1, \mathbf{d}}(\mathcal{S}_0))) + \lambda R_{b, \mathbf{d}} \right\}, \quad (4.25)$$

em que $D(\cdot)$ é uma função que mede a distorção, $\lambda > 0$ é o multiplicador de Lagrange, e $R_{b, \mathbf{d}}$ é a quantidade de *bits* gastos para indicar a direção selecionada. Note que $D(\cdot)$ deve refletir a distorção na imagem reconstruída. Como uma aproximação, $D(\cdot) = (\cdot)^2$ pode ser

usada como medida, assumindo-se que a maior parte dos coeficientes de alta frequência será quantizado para zero e que a distorção no domínio transformado será igual a nodomínio espacial (com a normalização apropriada). Também pode-se notar que $D(.) = (.)^2$ e $D(.) = |.|$ resultam em desempenho similar em codificação de vídeo [48]. Para a implementação adotada nesse trabalho, a medida de distorção foi adotada como descrito.

4.4.2 Correlação entre as direções de filtragem

Após a descrição da metodologia de implementação de uma (2-D DWT) direcional via *lifting* adotada, conclui-se que, para se completar um nível dessa transformada, seis direções devem ser escolhidas. Primeiramente, deve-se realizar o passo de predição entre as linhas, por exemplo, originando uma sub-banda de alta frequência H. Para tal, deve-se ter escolhido uma direção DP1 em (4.20), dentre os nove candidatos em (4.22). Em seguida, o primeiro passo de atualização será realizado, ao longo da direção escolhida DU1, obtendo-se uma sub-banda de baixa frequência L. Cada uma dessas sub-bandas passará por passos de predição e de atualização para se completar um nível de decomposição de uma DWT direcional, dessa vez, portanto, por meio de filtragens entre as colunas. Assim, a sub-banda L será decomposta em LH ao longo de DLP2 e, em LL ao longo de DLU2. Já a sub-banda H será decomposta em HH ao longo de DHP2 e, em HL ao longo de DHU2. A Figura 4.7 mostra apenas um esquemático dessas decomposições. Nota-se, então, que para se obter um nível completo de decomposição direcional têm-se 9⁶ possíveis combinações de direções. No entanto, é importante lembrar que as direcionalidades das imagens são localizadas, exigindo que as regiões em decomposição sejam suficientemente pequenas. Sugere-se a divisão mais simples: em blocos. Em princípio, é natural pensar que se poderia fazer um teste exaustivo, considerando, portanto, todas as possíveis combinações em cada bloco para escolher a ótima. Porém, deve-se notar que a decomposição de um bloco interfere na do outro. Assim, optou-se por realizar cada passo de *lifting* em todos os blocos da imagem, porém com direções individualizadas, antes de se passar para o próximo passo.

Observe, por ora, que existe uma grande complexidade na determinação da combinação ótima de direções. Além disso, na literatura, não é claro se existe qualquer correlação entre as direções de cada estágio. Frequentemente, opta-se por escolher, de maneira injustificada, a direção do passo de atualização igual à adotada para o passo de predição antecedente. Nesse trabalho, vários estudos e testes, que serão discutidos no Capítulo 5, foram realizados para elucidar essa questão central.

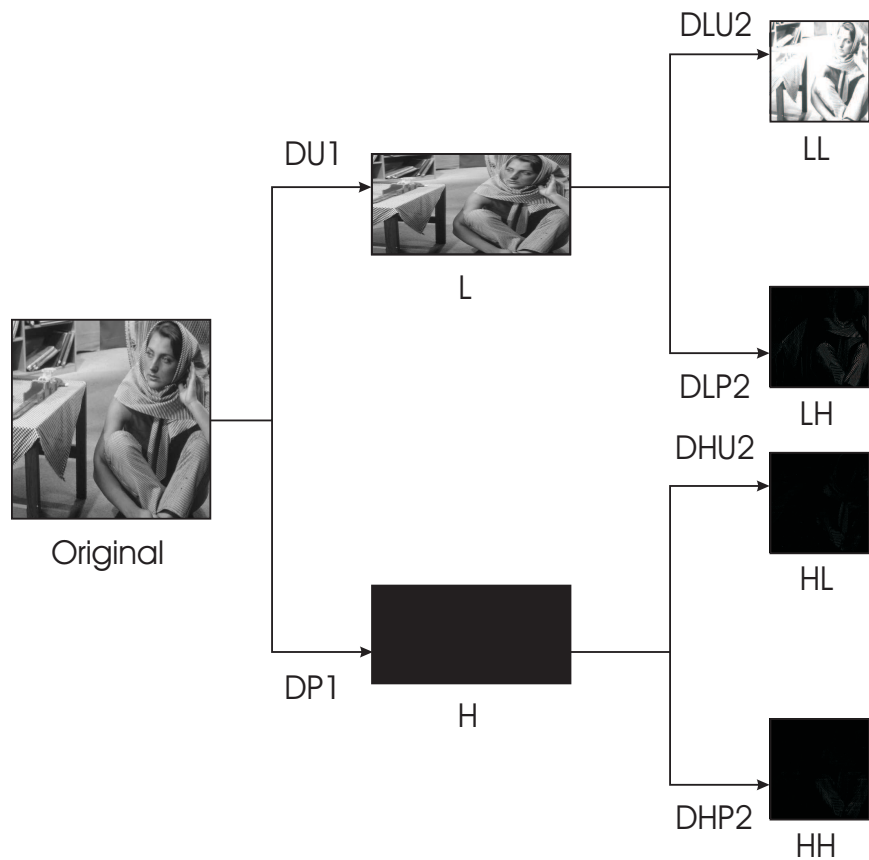


Figura 4.7: Esquemático da decomposição direcional em sub-bandas, considerando-se um nível completo de 2-D DWT. Os indicadores ao longo das setas referem-se às direções adotadas em cada passo de predição e de atualização realizado: DP1 - Direção de Predição 1; DU1 - Direção de *Update* 1; DLP2 - Direção de Predição da sub-banda *Low-pass* 2; DLU2 - Direção de *Update* da sub-banda *Low-pass* 2; DHP2 - Direção de Predição da sub-banda *High-pass* 2; DHU2 - Direção de *Update* da sub-banda *High-pass* 2.

Capítulo 5

Resultados

Nesse capítulo serão apresentados os resultados obtidos da implementação da transformada *wavelet* direcional, segundo a metodologia descrita no Capítulo 4, aqui chamada de D-DWT (do inglês, *Directional Discrete Wavelet Transform*). Para que se pudesse testar o método direcional nas imagens de interesse, ilustradas na Figura 5.1, os módulos de transformação de dois codificadores de imagem baseados em *wavelet* foram substituídos: o JPEG-2000 [49] e o SPIHT (do inglês, *Set Partitioning in Hierarchical Trees*) [50].

5.1 Validação

Inicialmente, apenas para se testar a adequação da substituição dos módulos de transformação, optou-se por realizar em separado a transformação padrão dos codificadores SPIHT e JPEG-2000, o banco cdf 9/7 [7], e, então, sobrescrever os coeficientes transformados originais. Nessa situação, os resultados obtidos a partir dos códigos modificados coincidiram com os resultados dos codificadores originais. Note que o JPEG-2000 apresenta uma alternativa a essa transformada, uma transformação inteira baseada no banco *spline* 5/3 [7], que não foi utilizada por se aplicar a casos de compressão sem perdas. Prosseguiu-se, então, à realização dos testes, propriamente ditos.

5.2 Implementação da transformada *wavelet* direcional (D-DWT)

Como visto no Capítulo 4, a transformada direcional implementada nesse trabalho é baseada na *wavelet* interpolante (6,6) [37]. A Seção 4.4.1, discute preliminarmente a

problemática da escolha das direções, informação difícil de ser apreendida dos artigos sobre o assunto. A proposta inicial desse trabalho era realizar a decomposição mais exaustiva possível. No entanto, durante os testes, notou-se que o fato de a decomposição de blocos vizinhos ser interdependente seria um óbice a essa estratégia. Prosseguiu-se, então, à inserção de restrições e à utilização de novos critérios de escolha.

5.2.1 Primeira abordagem: direção de atualização igual à de predição

A primeira abordagem consiste em impor a restrição de que a direção do passo de atualização seja a mesma do passo de predição que o antecede. Dessa maneira, o algoritmo de otimização baseado em distorção seleciona a direção de predição que a minimiza, que será herdada pelo passo de atualização. Além disso, observou-se que a decomposição adaptativa da sub-banda de alta frequência H em HH e HL poderia ser negligenciada sem muita perda no desempenho, visto que esses coeficientes já são bastante decorrelacionados, não apresentando orientações significativas a serem exploradas. Nesse caso, optou-se por sempre utilizar a direção $\check{\mathbf{d}}_0 = (1, 0)^T$. Aproveitando a oportunidade, justificativa semelhante pode ser apontada para o insucesso na decomposição direcional de resíduos de compensação de movimento. De forma similar, nos últimos níveis de transformação, $\mathbf{d}_0 = (1, 0)^T$ e $\check{\mathbf{d}}_0 = (1, 0)^T$ também eram sempre utilizadas. Assim, nos testes realizados, foram feitos 5 níveis de decomposição, no máximo, 3 desses eram direcionais. Quanto maior a quantidade de níveis, menor a quantidade de *pixels* em cada nova sub-banda, não justificando o gasto em que se incorreria para indicar as direções selecionadas.

Todas as imagens de teste possuem tamanho 512×512 , e foram divididas em blocos fixos de tamanho 16×16 . Os resultados apresentados ilustram curvas em que foram utilizados, em cada um dos codificadores: (i) 5 níveis de decomposição com a tradicional *wavelet* cdf 9/7; (ii) 5 níveis de decomposição com a tradicional *wavelet* interpolante (6,6); (iii) 3 níveis de decomposição utilizando a transformada direcional implementada (D-DWT) mais 2 níveis de decomposição com a tradicional *wavelet* interpolante (6,6). Os resultados estão ilustrados nas Figuras 5.2, 5.3, 5.4 e 5.5. As curvas sinalizadas com (S) foram codificadas pelo SPIHT, enquanto as sinalizadas com (J) foram codificadas pelo JPEG-2000.

A partir das curvas obtidas, nota-se que os resultados são muito bons para a imagem Bárbara. Essa imagem apresenta forte conteúdo direcional, principalmente no canto inferior direito em que se encontram listras. Já para as outras três imagens, nota-se que o método direcional equipara-se ao tradicional, visto que essas imagens já não apresentam conteúdo direcional localizado significativo. Diferentes artigos sobre o assunto, como, por exemplo, [37] e [46], reportam resultados semelhantes: o método direcional apresenta desempenho muito superior que o tradicional para um pequeno conjunto de imagens com

direcionalidades bastante localizadas. Infelizmente, os resultados aqui exibidos não puderam ser diretamente comparados aos expostos em [37], pois, mesmo utilizando igual conjunto de possíveis direções e igual banco de *wavelet*, o codificador utilizado pelos autores (o *TCE embedded bitplane coder*), não foi encontrado de forma disponível para testes. Uma questão interessante de se observar, motivada pelos resultados em [46], é que *wavelets* com deslocamentos menores (ou menor “pegada”), como a 5/3, parecem fornecer resultados ainda melhores. Note que, na D-DWT, a composição do banco direcional com os deslocamentos propostos permite que as funções de predição e de atualização utilizem vizinhos distantes até 15 *pixels* da amostra de interesse (ou uma pegada de 30 *pixels*), o que dificulta um pouco o aproveitamento das direcionalidades mais localizadas.

Considerando as restrições adotadas nessa primeira abordagem, propôs-se uma variação para se considerar, diretamente, a direção espacial da imagem como direção de predição. Nessa versão, a imagem original era submetida a um pré-processamento para ressaltar bordas e, em seguida, era realizada uma filtragem por um filtro de Gabor para que se pudesse identificar as direcionalidades espaciais. O conjunto de direções identificadas era utilizado como a primeira direção de predição de cada bloco, ao invés de se considerar o algoritmo de distorção. Os resultados se mostram piores do que os obtidos pelo algoritmo de minimização de distorção.

5.2.2 Segunda abordagem: novo critério de suavização para se escolher a direção de atualização independentemente

Sabe-se que a direção de predição determinará os coeficientes da sub-banda de alta frequência (H), enquanto a direção de atualização determinará os coeficientes da sub-banda de baixa frequência (L). A segunda abordagem se baseia nessa característica e propõe que a direção de predição continue sendo determinada pelo algoritmo de minimização de distorção e que a direção de atualização, porém, seja determinada por um novo critério que maximize a suavidade de L. Considerando que quanto mais suave, menor a variância, adotou-se como critério de maximização da suavidade a minimização da variância dos coeficientes da sub-banda L. Os resultados também se mostraram piores do que os da primeira abordagem, como se pode ver nas Figuras 5.6, 5.7, 5.8 e 5.9. Os resultados comparam decomposições com 1 nível direcional mais 5 níveis de *wavelet cdf 9/7*, com a tradicional decomposição em 6 níveis com a *cdf 9/7*, variando-se, porém, o tamanho de bloco. Observe que ambos os critérios são determinados no domínio transformado. Ora, dado que um bom resultado é caracterizado por boa qualidade de reconstrução (medida pela PSNR) a uma dada taxa (medida em *bits* gastos na representação), notou-se que o critério de distorção realmente assegurava taxas menores na codificação. Faltava, porém, analisar a qualidade de reconstrução das bases, visto que as curvas eram piores. Procedeu-se, então, à análise da estabilidade numérica da matriz polifásica para se entender melhor o que acontece com as bases da transformada à

medida em que as direções de atualização são escolhidas separadamente das de predição.

5.3 Análise da Estabilidade Numérica

A qualidade de reconstrução de um banco de filtros depende da inversibilidade de suas matrizes polifásicas. Uma maneira bastante utilizada para se analisar a sensibilidade ou a estabilidade de uma matriz, ou do sistema linear que ela representa, é através do cálculo de seu número de condição (ou em inglês, *condition number*)[51]. O número de condição de uma matriz \mathbf{A} quadrada, em relação a uma norma $\|\cdot\|$ é definido por:

$$\kappa(\mathbf{A}) = \|\mathbf{A}\| \|\mathbf{A}^{-1}\| \quad (5.1)$$

se \mathbf{A} é uma matriz não-singular; e $\kappa(\mathbf{A}) = +\infty$ se \mathbf{A} for singular.

Note que $\kappa(\cdot)$ depende da norma utilizada. Ao se utilizar, na definição acima, a norma L^2 , tem-se:

$$\kappa_2(\mathbf{A}) = \|\mathbf{A}\|_2 \|\mathbf{A}^{-1}\|_2 = \frac{\sigma_1(\mathbf{A})}{\sigma_n(\mathbf{A})}, \quad (5.2)$$

em que $\sigma_1(\mathbf{A})$ e $\sigma_n(\mathbf{A})$ são, respectivamente, o maior e o menor autovalor de \mathbf{A} . Matrizes com número de condição próximo de 1 são chamadas matrizes bem condicionadas, enquanto matrizes com número de condição muito maior que um (como, por exemplo, da ordem de 10^5), são chamadas matrizes mal condicionadas.

Desde já, pode-se perceber que a multiplicação da matriz \mathbf{A} por um escalar não altera seu número de condição. Seja:

$$\mathbf{A}\mathbf{v} = \sigma\mathbf{v}, \quad (5.3)$$

em que σ é um autovalor da matriz \mathbf{A} com autovetor correspondente \mathbf{v} . Pode-se multiplicar a matriz \mathbf{A} por um escalar k , obtendo-se:

$$(k\mathbf{A})\mathbf{v} = (k\sigma)\mathbf{v}. \quad (5.4)$$

Dessa maneira, note que cada um dos autovalores ficará multiplicado por k , não alterando a razão que representa o número de condição.

Considerando-se o exposto sobre o número de condição, pode-se aplicar essa análise de estabilidade numérica às matrizes polifásicas de síntese para investigar a baixa qualidade de reconstrução de algumas bases. Mais diretamente, considere que um sinal $\{x(n)\}$ seja quebrado em blocos e codificado após ser transformado por \mathbf{T} . Um dado bloco pode ser

representado pelo vetor \mathbf{x} , tal que seus coeficientes transformados sejam $\mathbf{y} = \mathbf{T}\mathbf{x}$. O sinal transformado será quantizado, resultando em $\hat{\mathbf{y}}$, que pode ser entendido como sendo a soma de \mathbf{y} com um ruído descorrelacionado \mathbf{r} , cuja variância seja proporcional à energia dos coeficientes transformados, isto é:

$$\sigma_{\mathbf{r}}^2 = \delta\sigma_{\mathbf{y}}^2, \quad (5.5)$$

em que $\sigma_{\mathbf{r}}^2$ e $\sigma_{\mathbf{y}}^2$ são, respectivamente, a variância do ruído e do bloco transformado, e δ é um escalar que representa fatores de normalização e de quantização. O sinal reconstruído $\hat{\mathbf{x}}$, pode, então, ser obtido por:

$$\hat{\mathbf{x}} = \mathbf{T}^{-1}(\mathbf{r} + \mathbf{y}) = \mathbf{T}^{-1}(\mathbf{r} + \mathbf{T}\mathbf{x}) = \mathbf{T}^{-1}\mathbf{r} + \mathbf{x}, \quad (5.6)$$

de maneira que o erro de reconstrução, representado por $\{\epsilon(n)\}$, seja:

$$\boldsymbol{\epsilon} = \hat{\mathbf{x}} - \mathbf{x} = \mathbf{T}^{-1}\mathbf{r}. \quad (5.7)$$

A variância do erro é dada por:

$$\sigma_{\boldsymbol{\epsilon}}^2 = E[\boldsymbol{\epsilon}^t \boldsymbol{\epsilon}] = E[\mathbf{r}^t \mathbf{T}^{-1,t} \mathbf{T}^{-1} \mathbf{r}] = E[\text{Tr} \{ \mathbf{T}^{-1} \mathbf{r} \mathbf{r}^t \mathbf{T}^{-1,t} \}] = E[\text{Tr} \{ \mathbf{T}^{-1} \mathbf{R}_{rr} \mathbf{T}^{-1,t} \}], \quad (5.8)$$

em que \mathbf{R}_{rr} é a matriz de auto-correlação de $\{r(n)\}$. Dado que $\{r(n)\}$ é não-correlacionado, tem-se $\mathbf{R}_{rr} = \delta\sigma_{\mathbf{y}}^2 \mathbf{I}$. Logo:

$$\sigma_{\boldsymbol{\epsilon}}^2 = \delta\sigma_{\mathbf{y}}^2 \text{Tr} \{ \mathbf{T}^{-1} \mathbf{T}^{-1,t} \} = \delta E[\mathbf{x}^t \mathbf{T}^t \mathbf{T} \mathbf{x}] \|\mathbf{T}^{-1}\| = \delta \text{Tr} \{ \mathbf{T} \mathbf{R}_{xx} \mathbf{T}^t \} \|\mathbf{T}^{-1}\| = \delta \text{Tr} \{ \mathbf{R}_{xx} \mathbf{T} \mathbf{T}^t \} \|\mathbf{T}^{-1}\|. \quad (5.9)$$

Assumindo-se que $\{x(n)\}$ é um sinal descorrelacionado, tem-se:

$$\sigma_{\boldsymbol{\epsilon}}^2 = \delta \|\mathbf{T}\| \|\mathbf{T}^{-1}\| = \delta \eta(\mathbf{T}), \quad (5.10)$$

em que $\eta(\cdot)$ é o número de condição de uma matriz.

Logo, nota-se que o erro de reconstrução é diretamente proporcional ao número de condição da matriz de transformação. Dessa maneira, transformações com número de condição maiores apresentarão erros de reconstrução também maiores. Pode-se, portanto, estender a modelagem desenvolvida para uma transformação linear \mathbf{T} para bancos de filtros unidimensionais, inicialmente. Nesse caso, como visto no Capítulo 3, tem-se um vetor $\mathbf{X}(z)$, com as transformadas das componentes polifásicas do sinal de entrada $\{x(n)\}$, e um vetor de saída $\mathbf{Y}(z)$, com os componentes passa-faixa resultantes da análise do sinal de entrada

por cada um dos M filtros $E_k(z)$, isto é, $\mathbf{Y}(z) = [y_0(z), y_1(z), \dots, y_{M-1}(z)]$, em que cada $y_k(z) = x_k(z)E_k(z)$, para $0 \leq k \leq M$. Matricialmente, $\mathbf{Y}(z) = \mathbf{X}(z)\mathbf{E}(z)$, em que $\mathbf{E}(z)$ é a matriz polifásica de análise. Assim, note que um banco de filtros é uma transformação linear para cada ponto $z = z_0$, de particular interesse em $z = e^{j\omega_0}$. Continuam sendo aplicáveis, portanto, as equações (5.5) a (5.10) com as modificações apropriadas. Nesse caso, em que se terão funções de z , o erro de reconstrução será descrito por:

$$\boldsymbol{\epsilon}(z) = \hat{\mathbf{x}}(z) - \mathbf{x}(z). \quad (5.11)$$

em que os vetores $\hat{\mathbf{x}}(z)$ e $\mathbf{x}(z)$ contêm, respectivamente, as polifásicas do sinal reconstruído e do sinal de entrada. Assim, tem-se que:

$$E[\boldsymbol{\epsilon}^t(z)\boldsymbol{\epsilon}(z)] = \delta\eta(\mathbf{E}(z)), \quad (5.12)$$

em que $\mathbf{E}(z)$ é a matriz polifásica de análise. Note que $\eta(\mathbf{T}) = \eta(\mathbf{T}^{-1})$, assim como $\eta(\mathbf{E}(z)) = \eta(\mathbf{R}(z))$, em que $\mathbf{R}(z)$ é a polifásica de síntese.

Observe que o número de condição é agora uma função de z e, por consequência, o erro de reconstrução também o será. Pode-se, portanto, avaliar a variação em frequência de ambos, fazendo-se $z = e^{j\omega}$:

$$E[\boldsymbol{\epsilon}^t(e^{j\omega})\boldsymbol{\epsilon}(e^{j\omega})] = \delta\eta(\mathbf{E}(e^{j\omega})). \quad (5.13)$$

Utilizando-se a notação desenvolvida na Seção 3.4 para bancos de filtros multidimensionais, pode-se, por fim, aplicar a modelagem descrita a sistemas dessa natureza. Nesse caso, tem-se:

$$E[\boldsymbol{\epsilon}^t(\mathbf{z})\boldsymbol{\epsilon}(\mathbf{z})] = \delta\eta(\mathbf{E}(\mathbf{z})). \quad (5.14)$$

Novamente, fazendo-se $\mathbf{z} = e^{j\boldsymbol{\omega}}$, vem:

$$E[\boldsymbol{\epsilon}^t(\boldsymbol{\omega})\boldsymbol{\epsilon}(\boldsymbol{\omega})] = \delta\eta(\mathbf{E}(\boldsymbol{\omega})). \quad (5.15)$$

Prossegue-se, então, à descrição detalhada da metodologia utilizada, desde o processo de composição da matriz polifásica polinomial a partir dos coeficientes da transformada inversa, até a avaliação do número de condição para diferentes frequências.

5.3.1 Metodologia

Para se realizar a análise da estabilidade numérica das matrizes polifásicas direcionais, realizou-se, para cada possível combinação de direções, a transformada inversa de uma

imagem transformada fictícia nula, com apenas um *pixel* não nulo em cada uma das sub-bandas por vez. A partir desse procedimento, foram obtidas as bases LL, LH, HL e HH referentes a cada combinação de direções. Dado que cada base tem suporte compacto, foi possível identificar a região não nula da imagem fictícia reconstruída, região essa que efetivamente representa a base de transformação. Considere, então, que os elementos dessa região de suporte tenham sido armazenados em uma matriz. Os elementos da matriz foram arranjados para compor as quatro submatrizes polifásicas de cada base. Para que se pudesse obter os polinômios em z de uma maneira consistente, com coeficientes dados pelas submatrizes, essas foram alinhadas em cima e à esquerda (ou, em inglês, *Top-left*). Por fim, cada um dos quatro polinômios passou a compor a linha correspondente à sua sub-banda na matriz polifásica, como mostrado na equação 3.102 do Capítulo 3. A Figura 5.10 mostra um esquemático do procedimento realizado, utilizando como exemplo o método aplicado à sub-banda LL.

Ao fim dessa parte do procedimento, obtém-se a matriz polifásica polinomial, isto é, uma matriz cujos elementos são polinômios em z_1 e em z_2 . Fez-se $z_1 = e^{j\omega_1}$ e $z_2 = e^{j\omega_2}$ e, variando-se cada uma das frequências de 0 a π , em dez pontos igualmente espaçados, pôde-se avaliar a variação em frequência do número de condição, para cada combinação de direções.

Para se fazer referência às direções consideradas em cada análise, serão utilizados os identificadores apresentados no final do Capítulo 4: DP1, DU1, DLP2, DLU2, DHP2 e DHU2. Além disso, apenas para caracterizar a direção adotada dentre as nove candidatas d_{-4} a d_4 , serão também atribuídos números inteiros, isto é, ao dizer que $DP1 = 1$, por exemplo, é equivalente a dizer que a direção de predição do primeiro estágio foi a d_{-4} . Destaca-se que a direção d_0 , horizontal ou vertical, dependendo do estágio de aplicação, recebe o indicador 5. A Tabela 5.1 ilustra a correspondência entre a notação adotada no Capítulo 4 e a que será utilizada nesse capítulo para designar as possíveis direções, além de mostrar o deslocamento na grade correspondente a cada uma delas.

Considere, como exemplo, a seguinte combinação de direções: DP1=5, DU1=5, DLP2=5, DLU2=5, DHP2=5 e DHU2=5. Nesse caso, que equivale à *wavelet* interpolante (6,6) separável, obteve-se a variação ilustrada na Figura 5.11. Já para DP1=8, DU1=5, DLP2=5, DLU2=5, DHP2=5 e DHU2=5, foi obtida a variação ilustrada na Figura 5.12. Nessas figuras, mostra-se, na parte de cima, um gráfico tridimensional das matrizes com os números de condição obtidos e, na parte de baixo, uma representação gráfica dessas matrizes. Observe que as linhas apresentam, de cima para baixo, variações em $\omega_2 = 0, 0.1\pi \dots, 0.9\pi$, enquanto as colunas apresentam, da esquerda para a direita, variações em $\omega_1 = 0, 0.1\pi \dots, 0.9\pi$. No anexo I encontram-se matrizes com os números de condição e suas representações gráficas para diversas combinações de direções.

Olhando-se as figuras do anexo I, pode-se notar que a base resultante da escolha de todas as direções iguais a 5, isto é, a combinação que representa a *wavelet* interpolante (6,6)

Tabela 5.1: Equivalência de notações

| Notação do Capítulo 4 | Notação para apresentação dos resultados | Deslocamento na grade |
|-----------------------|--|-----------------------|
| \mathbf{d}_{-4} | 1 | $(-3, 1)^T$ |
| \mathbf{d}_{-3} | 2 | $(-2, 1)^T$ |
| \mathbf{d}_{-2} | 3 | $(-1, 1)^T$ |
| \mathbf{d}_{-1} | 4 | $(-1, 3)^T$ |
| \mathbf{d}_0 | 5 | $(0, 1)^T$ |
| \mathbf{d}_1 | 6 | $(1, 3)^T$ |
| \mathbf{d}_2 | 7 | $(1, 1)^T$ |
| \mathbf{d}_3 | 8 | $(2, 1)^T$ |
| \mathbf{d}_4 | 9 | $(3, 1)^T$ |

separável, é a que apresenta maior estabilidade. Nesse caso, pode-se dizer que a matriz é bem condicionada em toda faixa de frequências. No entanto, ao se escolher a direção de predição diferente da de atualização, casos em que $DP1 \neq DU1$ ou $DLP2 \neq DLU2$, as bases apresentam números de condição maiores em faixas variadas do espectro.

Dessa maneira, dada uma combinação de direções, é possível que o número de condição varie de maneira abrupta em frequência. Isso é uma característica relevante, pois, mesmo que uma base seja extremamente biortogonal é possível que, simplesmente, se divida os coeficientes transformados pela energia de suas bases, normalizando-os de forma a não prejudicar o desempenho na codificação. No entanto, dada a abrupta variação verificada em várias das bases, esse procedimento não é suficiente para assegurar o bom desempenho, pois, já na introdução do conceito de número de condição, foi visto que esse procedimento de normalização não altera o número de condição de uma transformação. Seriam necessários, portanto, métodos mais complexos de compensação em frequência. Essa análise justifica o mau desempenho obtido em testes realizados com critérios independentes de escolha das direções de predição e de atualização em que os coeficientes transformados de cada sub-banda foram normalizados.

Em resumo, após os testes de análise da estabilidade numérica, pôde-se perceber que, dependendo da escolha das direções, as bases apresentam diferentes variações em frequência do número de condição, o que provoca condições ruins de reconstrução em certas combinações, especialmente quando as direções de predição são escolhidas independentemente das de atualização.



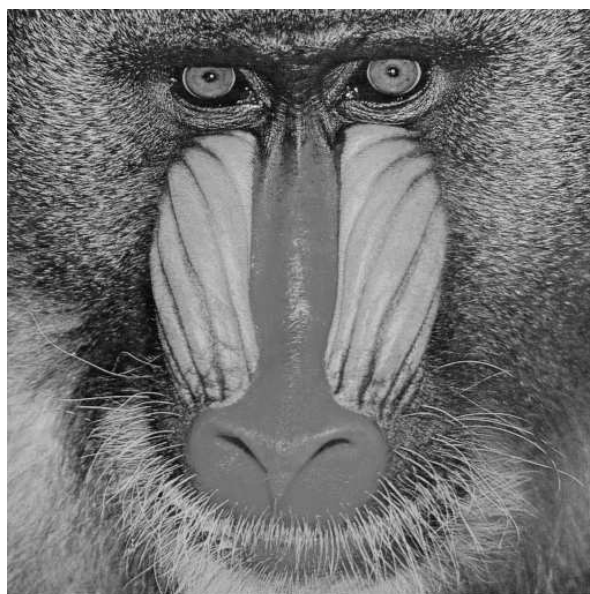
(a)



(b)



(c)



(d)

Figura 5.1: Imagens de Teste: (a) Barbara; (b) Lena; (c) GoldHill; (d) Baboon.

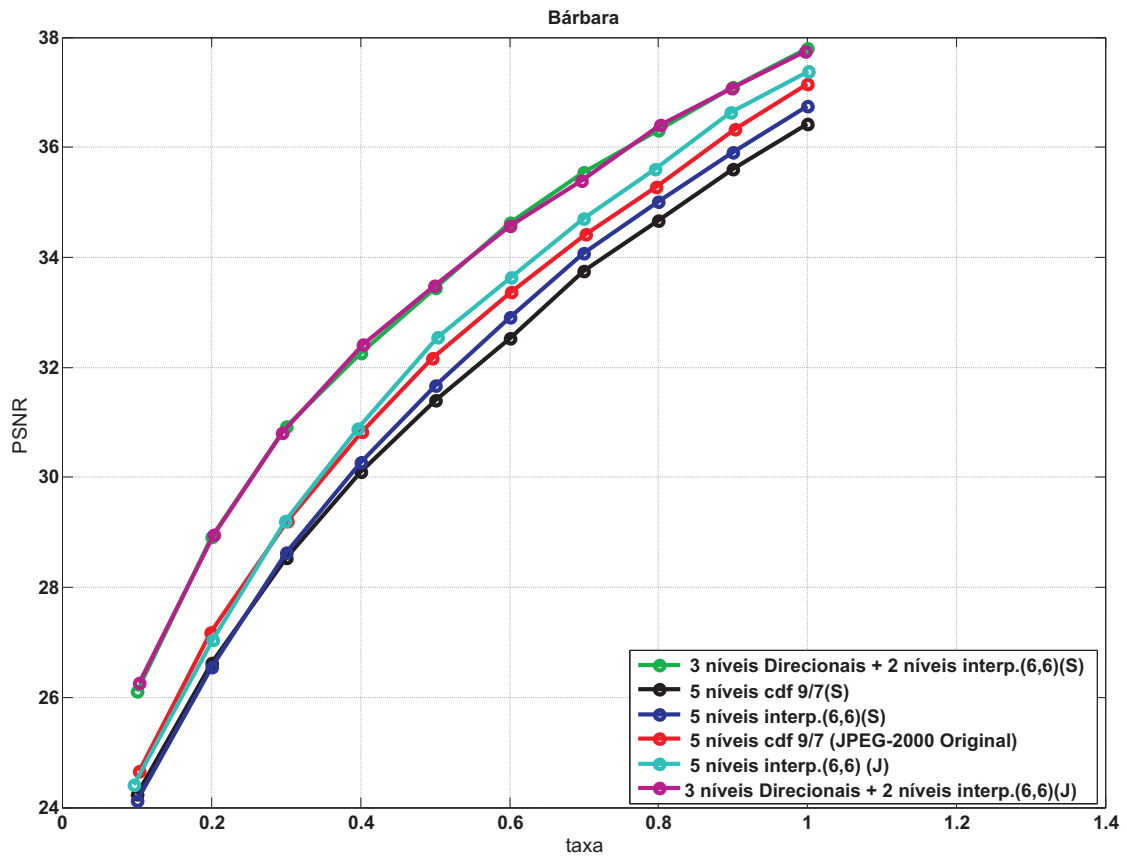


Figura 5.2: Curvas de $PSNR \times taxa$ resultantes da codificação da imagem Bárbara, utilizando-se a D-DWT com direções de predição e de atualização iguais.

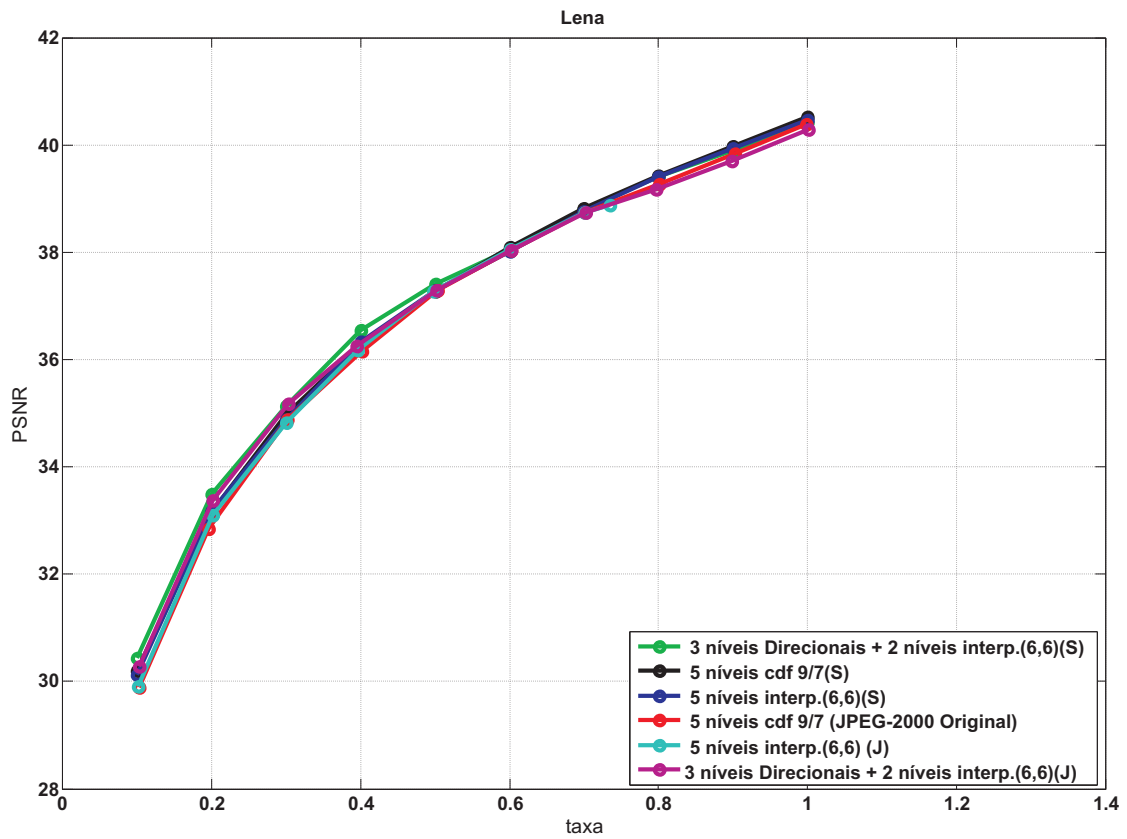


Figura 5.3: Curvas de $PSNR \times taxa$ resultantes da codificação da imagem Lena, utilizando-se a D-DWT com direções de predição e de atualização iguais.

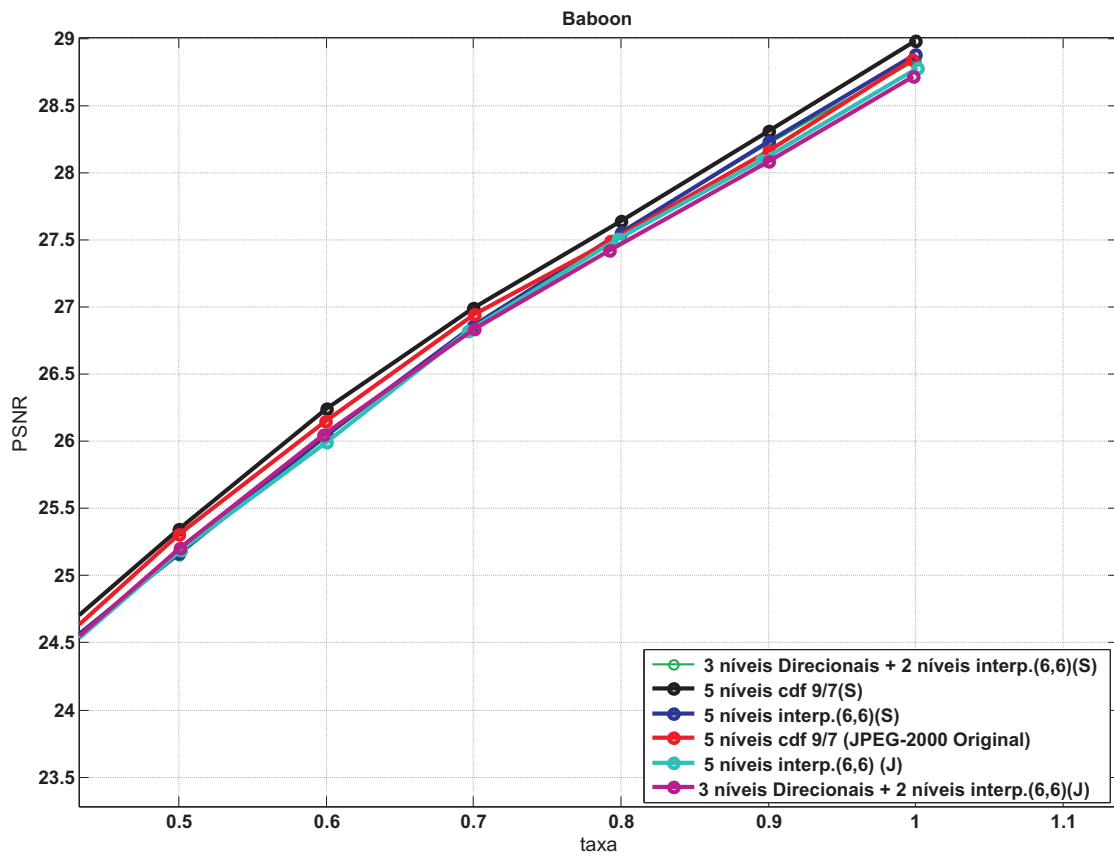


Figura 5.4: Curvas de $PSNR \times taxa$ resultantes da codificação da imagem Baboon, utilizando-se a D-DWT com direções de predição e de atualização iguais.

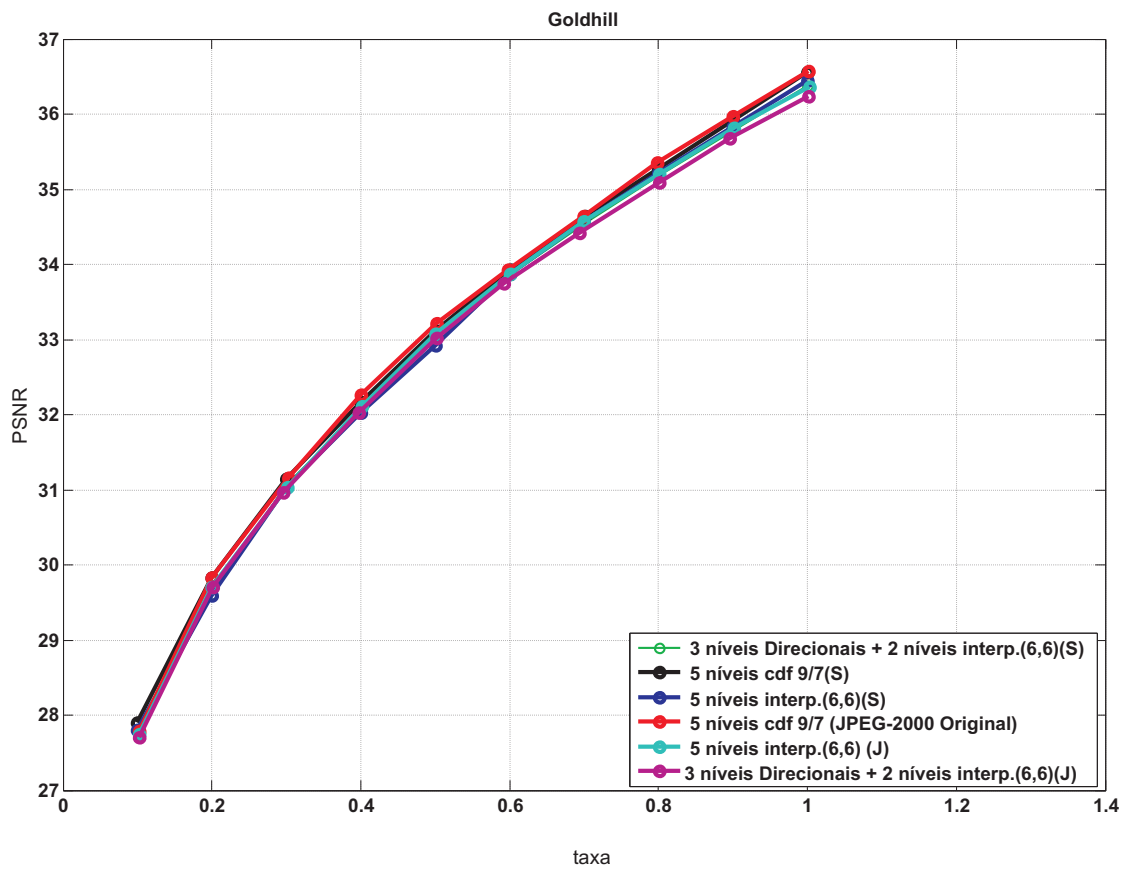
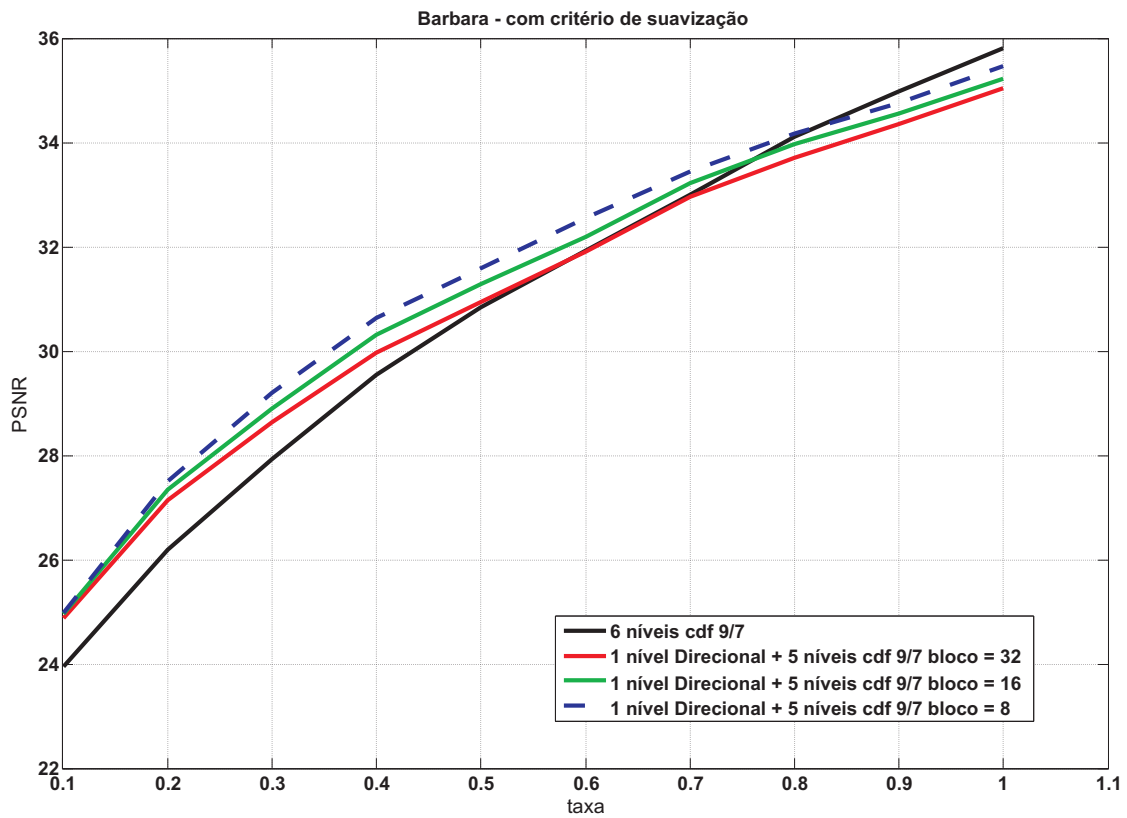
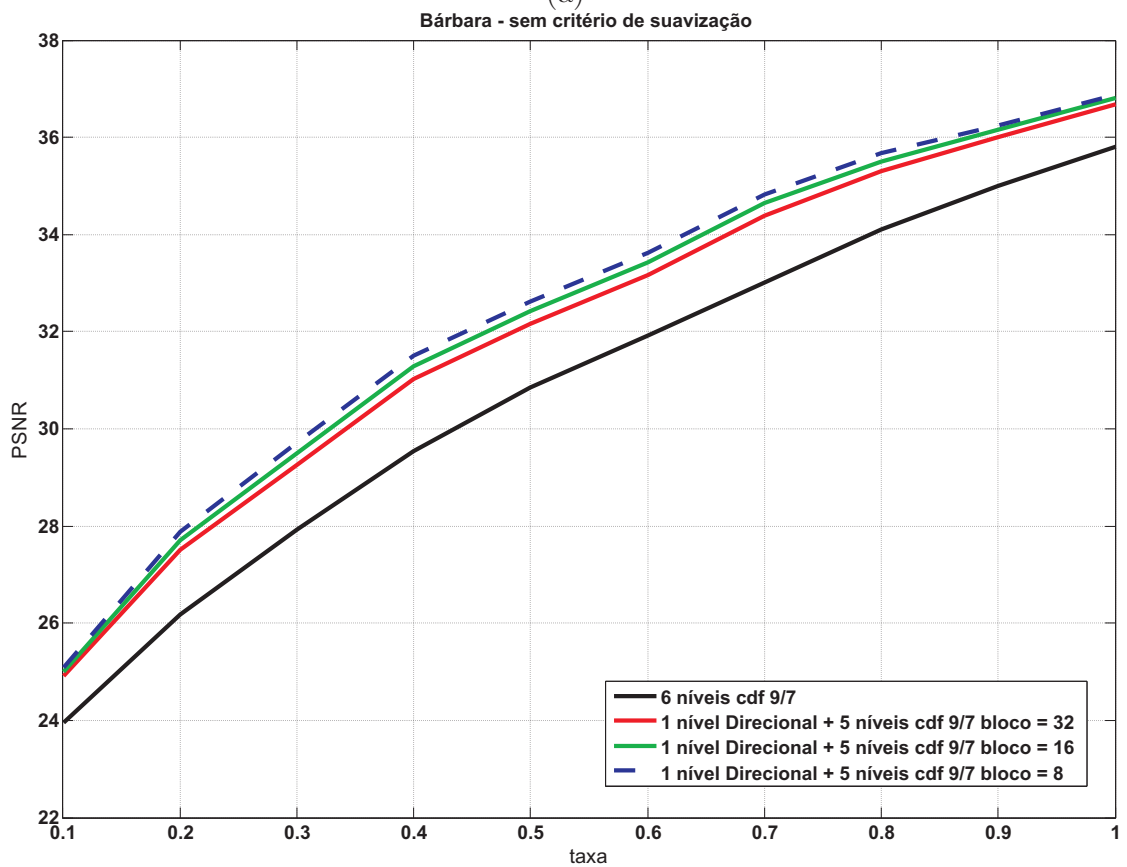


Figura 5.5: Curvas de $PSNR \times taxa$ resultantes da codificação da imagem Goldhill, utilizando-se a D-DWT com direções de predição e de atualização iguais.

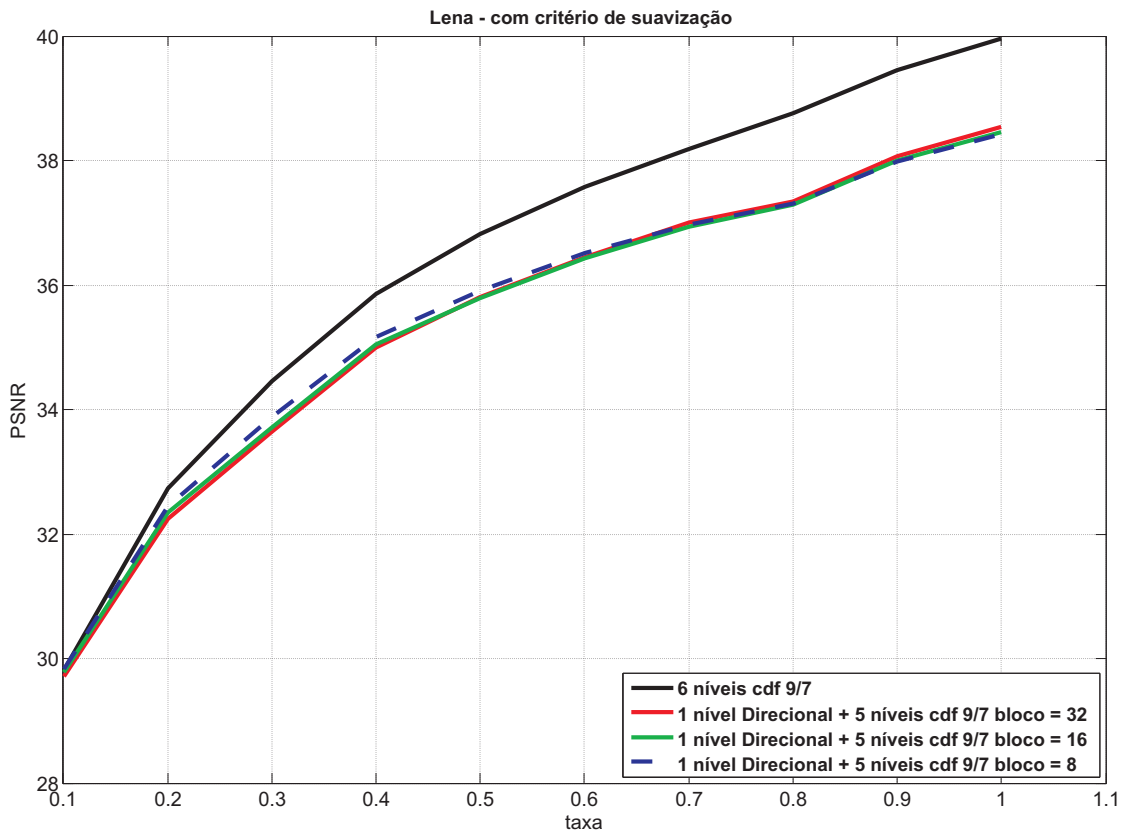


(a)

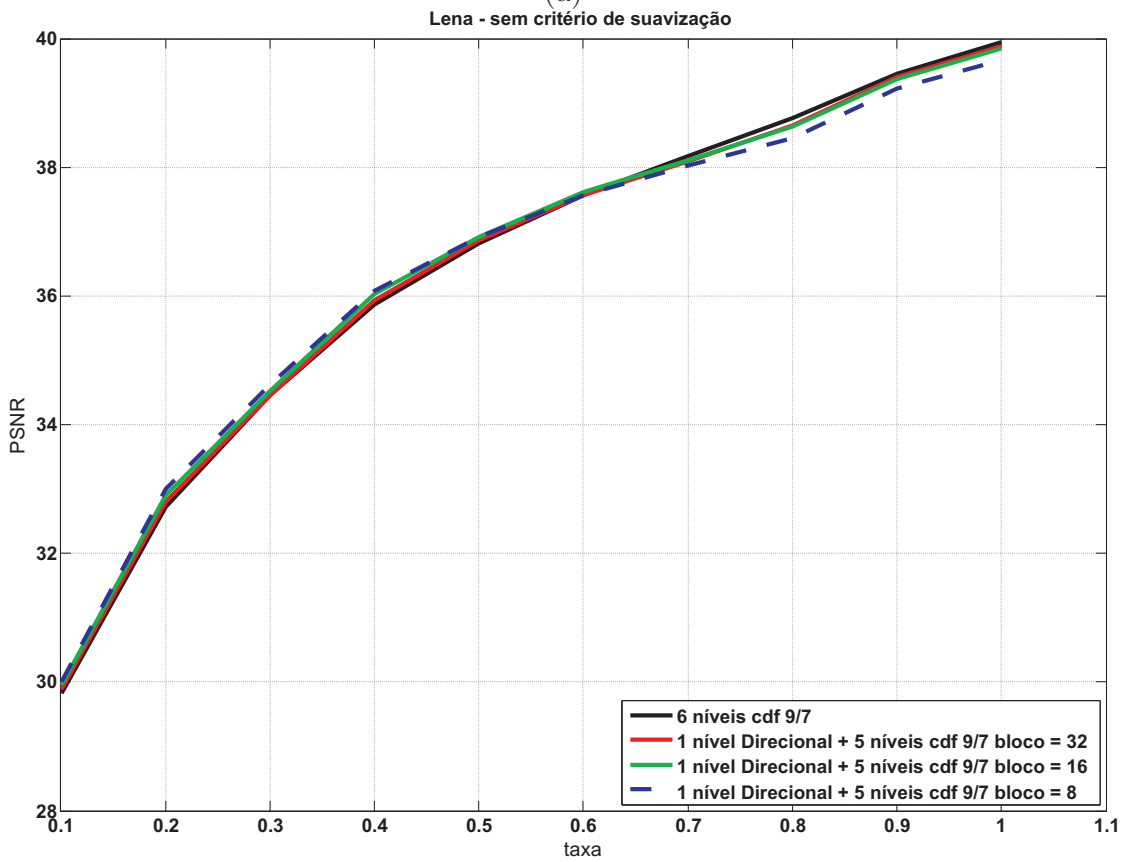


(b)

Figura 5.6: Testes de suavização - imagem Bárbara: (a) com critério de suavização; (b) sem critério de suavização.

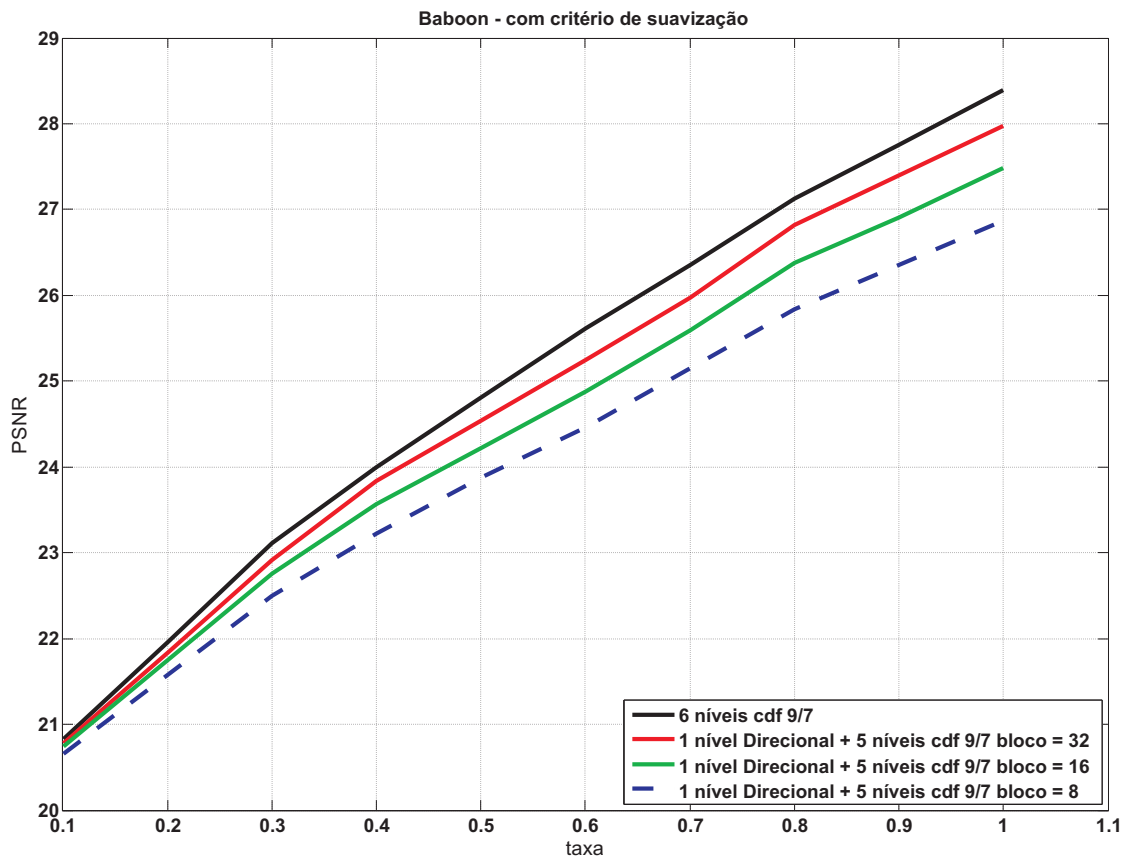


(a)

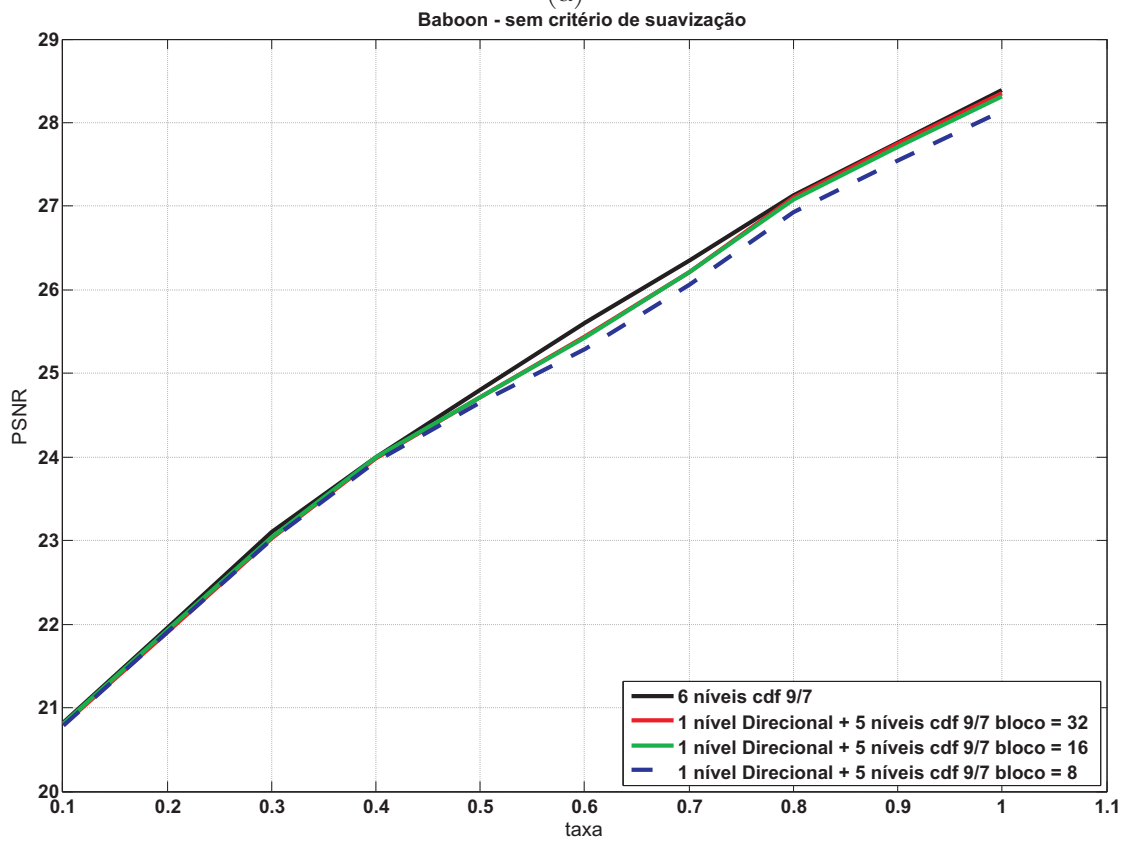


(b)

Figura 5.7: Testes de suavização - imagem Lena: (a) com critério de suavização; (b) sem critério de suavização.

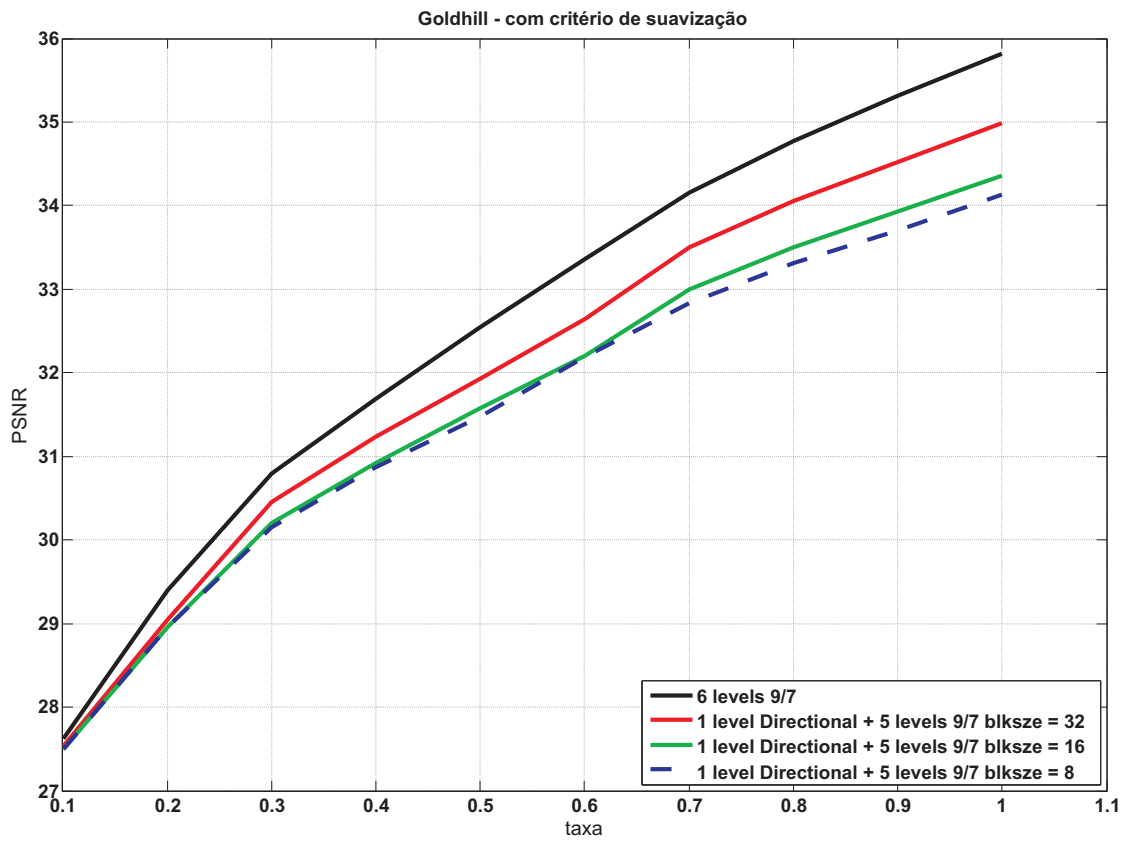


(a)

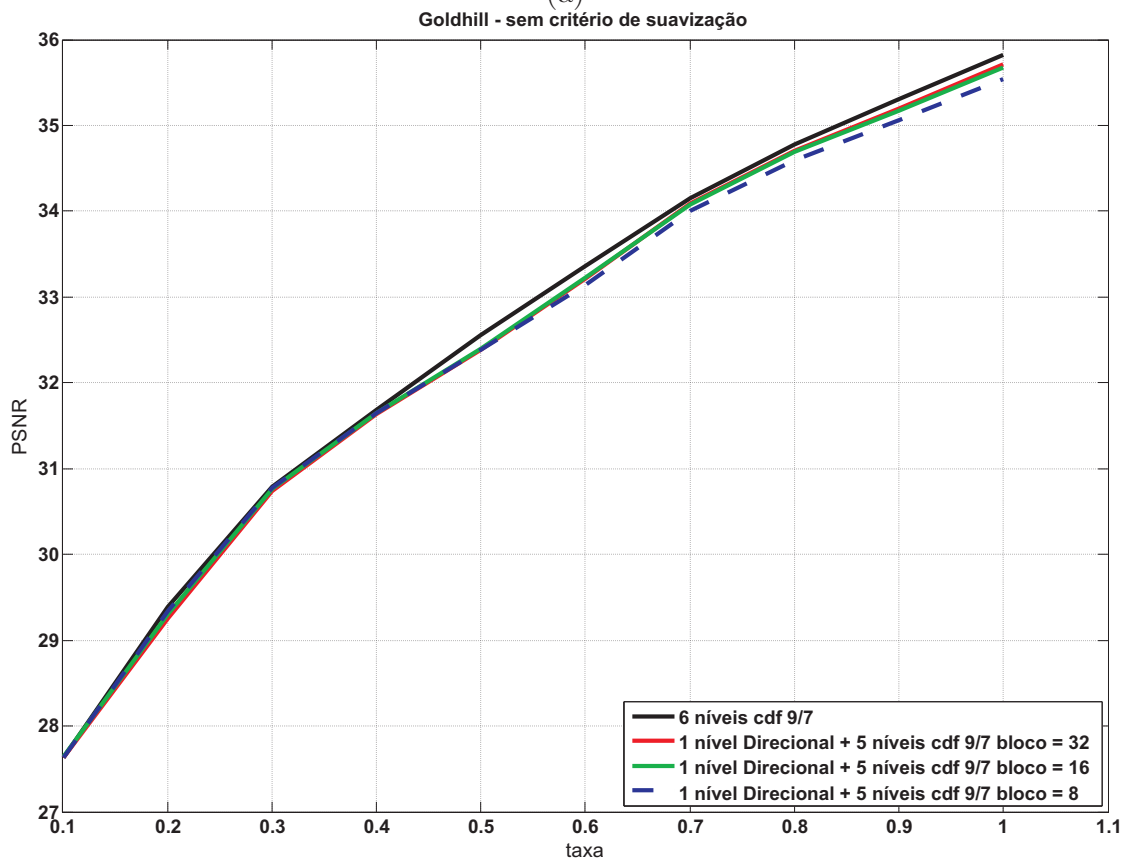


(b)

Figura 5.8: Testes de suavização - imagem Baboon: (a) com critério de suavização; (b) sem critério de suavização.



(a)



(b)

Figura 5.9: Testes de suavização - imagem Goldhill: (a) com critério de suavização; (b) sem critério de suavização.

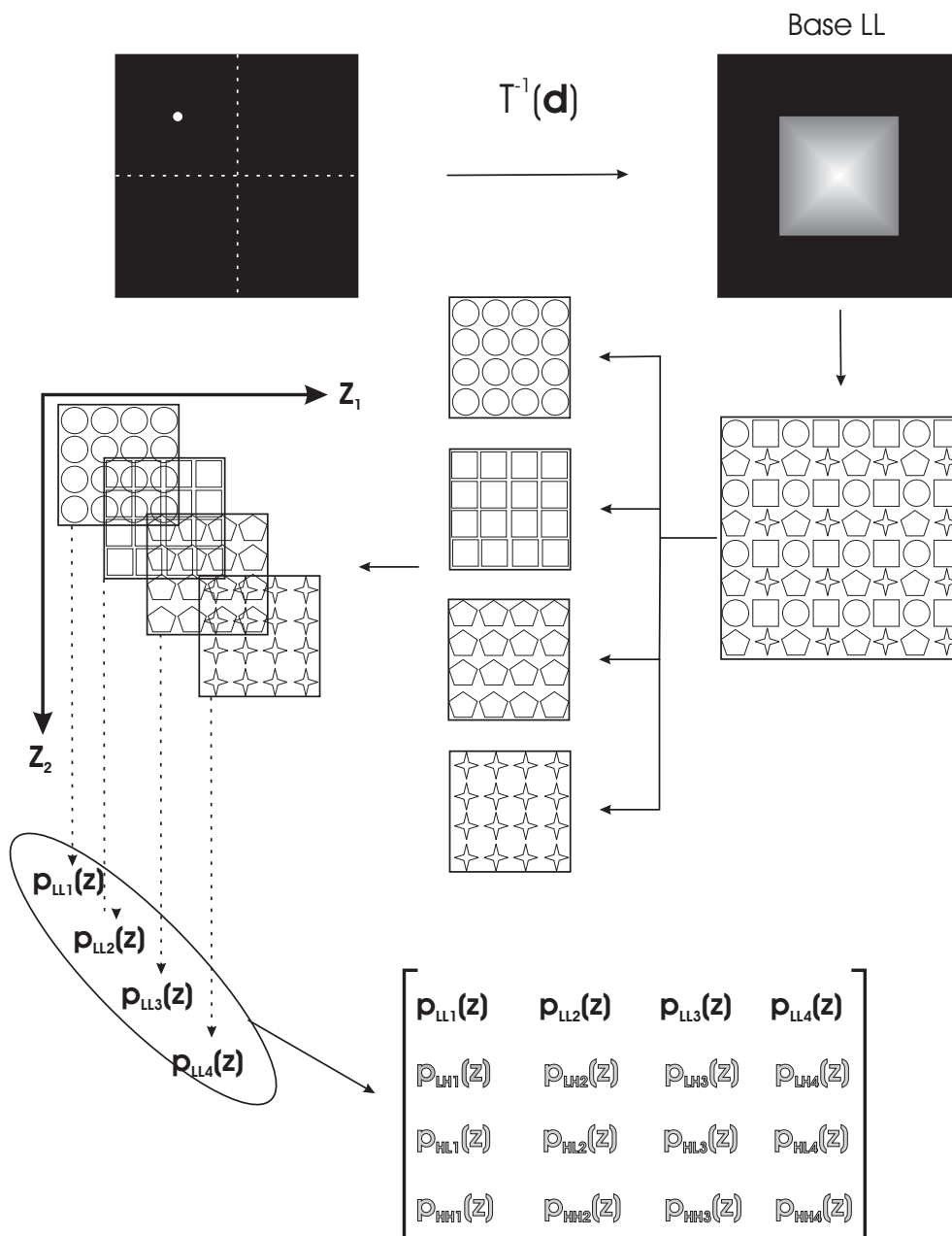


Figura 5.10: Esquemático da metodologia de composição da matriz polifásica de síntese a partir dos coeficientes da transformada inversa, considerando a aplicação à sub-banda LL como exemplo.

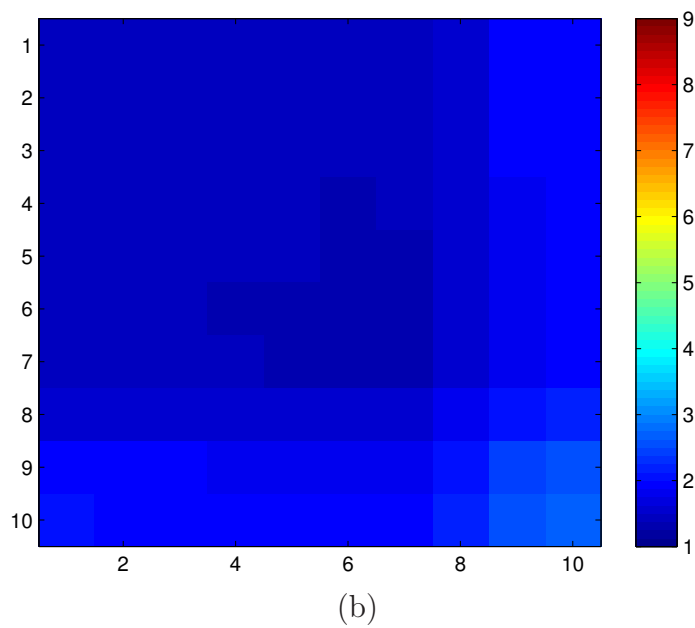
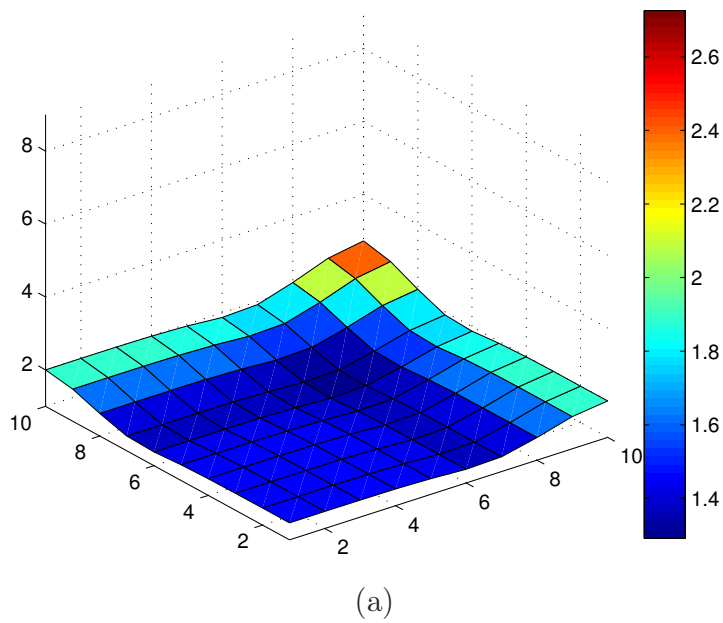
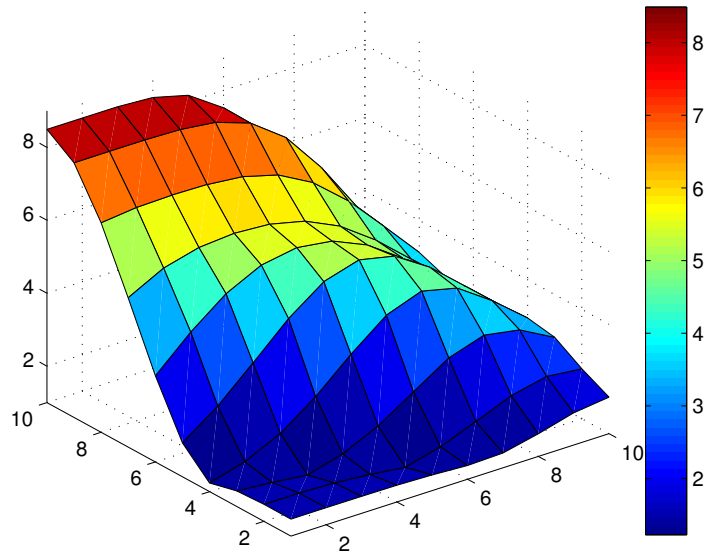
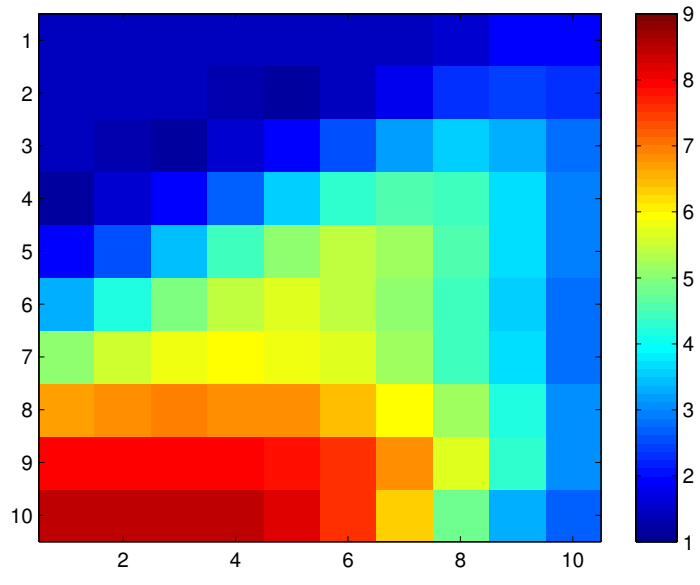


Figura 5.11: Variação do número de condição, $\eta(\mathbf{E}(\omega_1, \omega_2))$, referente à *Wavelet* interpolante (6,6) convencional: $DP1 = 5$, $DU1 = 5$, $DLP2 = 5$, $DLU2 = 5$, $DHP2 = 5$ e $DHU2 = 5$. (a) Representação tridimensional, onde ω_1 e ω_2 são os eixos horizontais. (b) Representação bidimensional a cores da variação de $\eta(\mathbf{E}(\omega_1, \omega_2))$, contendo ω_1 e ω_2 nas linhas e colunas.



(a)



(b)

Figura 5.12: Variação do número de condição, $\eta(\mathbf{E}(\omega_1, \omega_2))$, referente à combinação de direções $DP1 = 8$, $DU1 = 5$, $DLP2 = 5$, $DLU2 = 5$, $DHP2 = 5$ e $DHU2 = 5$, contendo, portanto, direção de predição diferente da de atualização. (a) Representação tridimensional, onde ω_1 e ω_2 são os eixos horizontais. (b) Representação bidimensional a cores da variação de $\eta(\mathbf{E}(\omega_1, \omega_2))$, contendo ω_1 e ω_2 nas linhas e colunas.

Capítulo 6

Conclusões

6.1 Sumário do trabalho e avaliação dos resultados

O objetivo principal do trabalho foi desenvolver uma metodologia adequada para a análise da estabilidade numérica de transformadas *wavelet* direcionais adaptativas implementadas via *lifting*, buscando melhorar o desempenho na codificação de imagens. Para tal, procedeu-se à implementação de uma transformada dessa natureza baseada no banco interpolante (6,6).

A estrutura *lifting* de implementação nada mais é do que uma maneira de se decompor uma matriz polifásica, portanto, polinomial, em um produto de matrizes elementares. A estrutura *lifting*, baseada em estágios canônicos obtidos através da decomposição de Smith, é um jeito eficiente de se implementar uma transformada *wavelet*, em que a resposta impulsional de cada filtro FIR pode ser fatorada em alguns estágios. Popularmente, esses estágios são denominados separação (ou, em inglês, “Split”), predição (ou, em inglês, “Prediction”) e atualização (ou, em inglês, “Update”). O passo de separação divide as amostras de um sinal em pares e ímpares. O passo de predição se utiliza das amostras pares, por exemplo, para resultar em um bom preditor para cada uma das amostras ímpares. O resíduo entre a amostra e seu valor predito é chamado de detalhe. Já o passo de atualização se utiliza dos coeficientes de detalhe do conjunto ímpar para atualizar as amostras pares.

O método adotado nesse trabalho para a inserção da informação direcional na transformada se baseia na alteração da vizinhança considerada nos operadores de predição e de atualização, sendo, portanto, considerado um método com foco espacial e não frequencial. Dessa maneira, cada direção selecionada é capaz de alterar o domínio das funções de predição e de atualização. No entanto, uma informação difícil de ser apreendida dos artigos sobre o assunto é como as direções dos passos sucessivos podem ser escolhidas de maneira ótima, isto é, de maneira a garantir uma melhor qualidade na reconstrução de

uma imagem, objeto central do trabalho, a uma dada taxa. Identifica-se, então, a grande problemática: como escolher as direções dos passos de *lifting* para oferecer uma melhoria no desempenho da codificação de uma imagem. Para responder a essa questão, buscou-se uma métrica adequada para encontrar critérios e restrições capazes de determinar que tipo de escolha seria capaz de promover o melhoramento desejado.

Primeiramente, adotou-se um critério de minimização de distorção, no domínio transformado, capaz de determinar as direções de predição. Decidiu-se por, simplesmente, fazer com que o passo de atualização herdasse a direção de predição. Os resultados obtidos com essa estratégia muito se assemelham aos mostrados em artigos recentes sobre o assunto: um pequeno conjunto de imagens com direcionalidades fortemente localizadas apresentam desempenho muito superior quando decompostas por métodos direcionais, se comparados ao método tradicional separável.

Sabendo-se que o passo de atualização origina a sub-banda de baixa frequência, seria razoável supor que critérios de maximização da suavidade dessa sub-banda oferecessem melhor desempenho. No entanto, ao escolher as direções dos dois passos de maneira independente, foram obtidos resultados piores do que os obtidos com direções de predição e de atualização iguais. Adotou-se, então, uma métrica capaz de evidenciar aspectos afetos à reconstrução ou, de forma equivalente, à inversibilidade das transformadas direcionais: o número de condição.

Desenvolveu-se, portanto, uma nova metodologia para avaliar a qualidade de reconstrução das transformadas após a inserção do ruído de quantização, fundamentada no número de condição. Considerando que uma matriz polifásica representa o banco de filtros do qual se originou, decidiu-se utilizar esse escalar, bastante difundido em análise de estabilidade de sistemas, para avaliar a sensibilidade do banco de síntese para cada conjunto de direções arbitrárias escolhidas. Constatou-se que o caso tradicional separável é bem condicionado em toda a faixa de frequências. Igual comportamento pôde ser verificado nos casos direcionais em que as direções de predição e de atualização eram iguais. Diferentemente, portanto, foram os casos em que essas direções foram escolhidas independentemente. Nesses casos, as variações em frequência do número de condição se mostraram abruptas, restando prejudicada a inversibilidade da transformada correspondente, considerando a injeção do ruído de quantização no processo.

6.2 Sugestão para trabalhos futuros

Os resultados obtidos nesse trabalho apontam para a adequação do método implementado na representação de imagens com fortes direcionalidades localizadas. No entanto, dada a importância em compressão de imagens e, portanto, em compressão de vídeo de se extrair o conteúdo direcional de uma cena, sugerem-se as seguintes contribuições para trabalhos

futuros:

- Considerando-se a problemática da escolha do conjunto ótimo de direções no contexto *lifting*, sugere-se o desenvolvimento de métodos adequados de compensação em frequência para aumentar a qualidade de reconstrução de transformadas em que as direções de predição e de atualização tenham sido escolhidas de maneira independente, ampliando a quantidade de escolhas competitivas;
- Ainda considerando o tipo de abordagem utilizada nesse trabalho para a implementação de transformadas direcionais, aconselha-se o desenvolvimento analítico, a partir dos operadores direcionais de predição e de atualização, da matriz polifásica de síntese para que se tenha um entendimento ainda maior sobre a influência da correlação entre as direções na qualidade de reconstrução da transformada, bem como a sua relação com a característica de biortogonalidade;
- Considera-se igualmente importante comparar os resultados obtidos com a rotação do banco da interpolante (6,6) com outros bancos conhecidos e com menores pegadas como, por exemplo, os bancos da cdf 9/7 e da *spline* 5/3;
- É cabível uma comparação entre métodos cujas possíveis direções de rotação do banco restringem-se a deslocamentos inteiros com outros métodos em que abordagens *sub-pixel* são admitidas;
- Encoraja-se, por fim, a comparação da eficiência na codificação de imagens e de vídeos entre métodos baseados em rotação de bancos de *wavelets*, como o utilizado nesse trabalho, métodos de transformadas direcionais de bloco (como a DCT), e métodos puramente espaciais de predição intra-quadros.

REFERÊNCIAS BIBLIOGRÁFICAS

- [1] INTERNATIONAL TELECOMMUNICATIONS UNION. *ITU-T T.81 - Information Technology - Digital Compression and Coding of Continuous-Tone Still Images - Requirements and Guidelines*. Setembro 1992.
- [2] INTERNATIONAL TELECOMMUNICATIONS UNION. *ITU-T Recommendation H.262 : Information technology - Generic coding of moving pictures and associated audio information: Video*. Julho 1995.
- [3] SWELDENS, W. The lifting scheme: A construction of second generation wavelets. *SIAM J. Math. Anal.*, v. 29, 1998.
- [4] DAUBECHIES, I.; SWELDENS, W. Factoring wavelet transforms into lifting steps. *J. Fourier Anal. Appl.*, v. 4, 1998.
- [5] PENNEBAKER, W. B.; MITCHELL, J. L. *JPEG: Still Image Compression Standard*. Second. [S.l.]: Van Nostrand Reinhold, 1993.
- [6] QUEIROZ, R. L. de. Compression of color images. In: *Handbook on Transforms and Data Compression*, G. Sharma. [S.l.]: Ed. Boca Raton.
- [7] TAUBMAN, D. S.; MARCELLIN, M. M. *JPEG2000:Image Compression Fundamentals, Standards, and Practice*. Second. [S.l.]: Springer, 2001.
- [8] KOVACEVIC, J.; VETTERLI, M. Nonseparable multidimensional perfect reconstruction filter banks and wavelet bases for rn. *IEEE Transactions on Information Theory*, v. 38 no. 2, 1992.
- [9] COHEN, A.; DAUBECHIES, I. Nonseparable bidimensional wavelet bases. *Rev. Mat. Iberoamericana*, v. 9 no. 1, 2008.
- [10] VAIDYANAYHAN, P. *Multirate Systems and Filter Banks*. Second. [S.l.]: Prentice-Hall Signal Processing Series, 2001.
- [11] OPPENHEIM R.W. SCHAFFER, J. B. A. *Discrete-Time Signal Processing*. 2.ed. ed. Upper Saddle River, NJ, EUA: Prentice-Hall, 1997. ISBN 0-13-814757-4.

- [12] LATHI, B. P. *Modern Digital and Analog Communication Systems*. 3.ed. ed. New York, NY, USA: Oxford University Press, Inc., 1998. ISBN 0195110099.
- [13] CROCHIERE, R. E.; RABINER, L. R. *Digital Image Processing*. Second. [S.l.]: Prentice Hall, 2001.
- [14] DINIZ E. A. B. DA SILVA, S. L. N. P. S. R. *Processamento digital de sinais: projeto e análise de sistemas*. Porto Alegre: Bookman, 2004. 590p. p. ISBN 85-363-0418-9.
- [15] CROISIER, D. E. A.; GALAND, C. Perfect-channel splitting by use of interpolation/decimation tree decomposition technique. *Int. Symp. on Info., Circuits and Systems*.
- [16] VAIDYANATHAN, P. P. Theory and design of m-channel maximally decimated quadrature mirror filters with arbitrary m, having perfect reconstruction property. *IEEE Trans. on Acoustics, Speech and Signal Processing*, ASSP-35.
- [17] VAIDYANATHAN, P. P. Quadrature mirror filter banks, m-band extensions and perfect-reconstruction techniques. *IEEE ASSP magazine*, v. 4.
- [18] VAIDYANATHAN, P. P. How to capture all fir perfect reconstruction qmf banks with unimodular matrices? *Proc. IEEE Symp. Circuits and Sys*.
- [19] CHEN, T.; VAIDYANATHAN, P. P. Recent developments in multidimensional multirate systems. *IEEE Transactions on CSVT*, v. 3 no. 2, 1993.
- [20] VETTERLI, M. Wavelets, approximation and compression. *IEEE Signal Processing Magazine*, Sep., 2001.
- [21] CANDÈS, E. J.; DONOHO, D. L. Ridgelets: a key to higher-dimensional intermittency? *Phil. Trans. R. Soc. Lond. A*, 1999.
- [22] CANDÈS, E. J.; DONOHO, D. L. Curvelets — a surprisingly effective nonadaptive representation for objects with edges. *A. Cohen, C. Rabut, and L. L. Schumaker, editors, Curve and Surface Fitting, Saint-Malo*, 1999.
- [23] DAUGMAN, J. Two-dimensional spectral analysis of cortical receptive field profile. *Vision Research*, 1980.
- [24] OLSHAUSEN, B. A.; FIELD, D. J. Emergence of simple-cell receptive field properties by learning a sparse code for natural images. *Nature*, v. 381, 1996.
- [25] BAMBERGER, R. H.; SMITH, M. J. T. A filter bank for the directional decomposition of images: Theory and design. *Nature*, v. 40 no. 4, 1992.
- [26] WIEGAND G. J. SULLIVAN, G. B. T.; LUTHRA, A. Overview of the h.264/avc video coding standard. *IEEE Trans. Circuits Syst. Video Technol.*, v. 13 no. 7, 2003.

- [27] IKONOMOPOULOS, A.; KUNT, M. High compression image coding via directional filtering. *Signal Processing*, v. 3 no. 4, 1985.
- [28] LI, H.; HE, Z. Directional subband coding of images. *Proc. ICASSP*, v. 3, 1989.
- [29] BAMBERGER, R. H.; SMITH, M. A filter bank for the directional decomposition: Theory and design. *IEEE Trans. Signal Process.*, v. 40 no. 4, 1992.
- [30] CANDÈS, E. J.; DONOHO, D. L. Monoscale ridgelets for the representation of images with edges. *Tec. Rep., Dept. Statist., Stanford Univ., Stanford, CA*, 1999.
- [31] DO, M. N.; VETTERLI, M. The finite ridgelet transform for image representation. *IEEE Trans. Image Process.*, v. 12 no. 1, 2003.
- [32] MEYER, F. G.; COIFMAN, R. R. Brushlets: a tool for directional image analysis and image compression. *J. Appl. Comput. Harmon. Anal.*, v. 5, 1997.
- [33] GONZALEZ, R. C.; WOODS, R. E. *Processamento de imagens digitais*. Second. [S.l.]: Edgard Blucher, 2000.
- [34] TAUBMAN, D.; ZAKHOR, A. Orientation adaptive subband coding of images. *IEEE Trans. Image Process.*, v. 3 no. 4, 1994.
- [35] SWELDENS, W. The lifting scheme: A custom-design construction of biorthogonal wavelets. *Appl. Comput. Harmon. Anal.*, v. 3, n. 2, p. 186–200, 1996.
- [36] CHANG, A. M. C.-L.; GIROD, B. Adaptive wavelet transform for image compression via directional quincunx lifting. *Proc. IEEE International Workshop on Multimedia Signal Processing, MMSP 2005, Shanghai, China*, 2005.
- [37] CHANG, C.-L.; GIROD, B. Direction-adaptive discrete wavelet transform via directional lifting and bandeletization. *Proc. IEEE International Conference on Image Processing, Atlanta, GA*, 2006.
- [38] CHANG, C.-L.; GIROD, B. Direction-adaptive discrete wavelet transform for image compression. *IEEE International Conference on Image Processing*, v. 16 no.5, 2007.
- [39] DING, F. W. W.; LI, S. Lifting-based wavelet transform with directionally spatial prediction. *Picture Coding Symp., San Francisco, CA*, Dec. 2004.
- [40] DING F. WU, X. W. S. L. W.; LI, H. Adaptive directional lifting-based wavelet transform for image coding. *IEEE Trans. Image Process.*, v. 16 no. 2, Feb. 2007.
- [41] CALDERBANK, R. et al. Wavelet transforms that map integers to integers. *Appl. Comput. Harmon. Anal.*, v. 5, 1998.

- [42] JAIN, A. K.; FARROKHNIYA, F. Unsupervised texture segmentation using gabor filters. *Pattern Recognition*, v. 24, 1991.
- [43] JAIN, A. K. *Fundamentals of Digital Image Processing*. [S.l.]: Prentice Hall, 1989.
- [44] JAYANT, N. S.; NOLL, P. *Digital coding of waveforms*. [S.l.]: Prentice Hall, Englewood Cliffs, NJ, 1984.
- [45] VETTERLI, M.; HERLEY, C. Wavelets and filter banks: Theory and design. *IEEE Trans. Acoust. Speech Signal Process.*, v. 40 no. 9, 1992.
- [46] LIU, Y.; NGAN, K. Weighted adaptive lifting-based wavelet transform for image coding. *IEEE International Conference on Image Processing*, v. 17, 2008.
- [47] GIROD, B. Rate-constrained motion estimation. in *Proc. SPIE Visual Communications and Image Processing*, Chicago, IL, 1994.
- [48] WIEGAND, T.; GIROD, B. Lagrange multiplier selection in hybrid video coder control. in *Proc. IEEE Int. Conf. Image Process.*, v. 3, Thessalonik, Greece, Oct. 2001.
- [49] ISO/IEC 15444-1. JPEG2000 image coding system. 2000.
- [50] SAID, W. A. P. A. A new, fast, and efficient image codec based on set partitioning in hierarchical trees. *IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY*, v. 6, 1996.
- [51] GOLUB, G. H.; LOAN, C. F. V. *Matrix Computations*. Third. [S.l.]: The Johns Hopkins University Press, 1996.

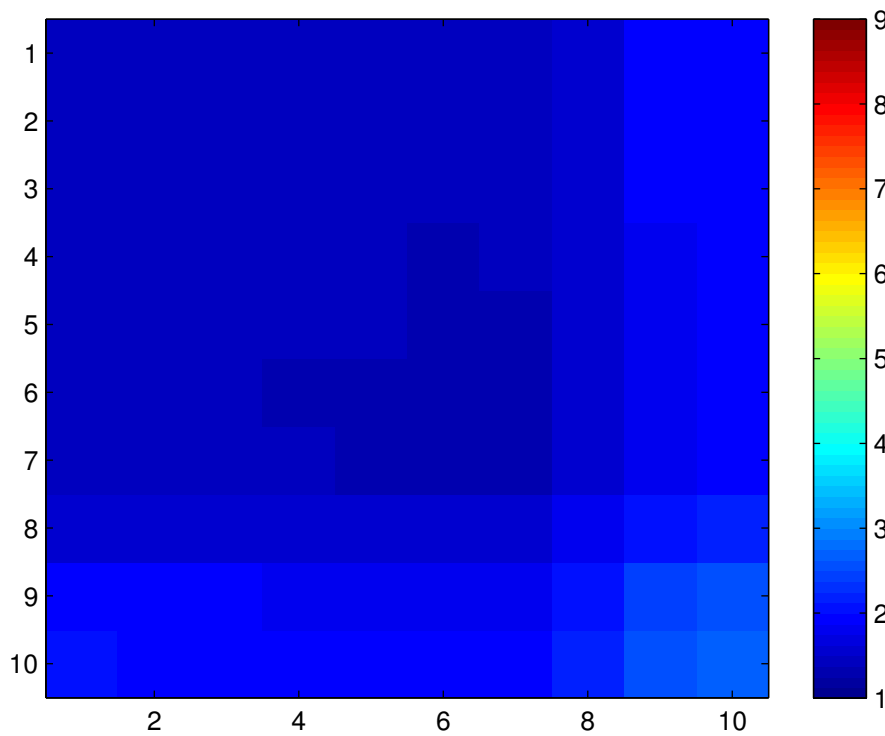
ANEXOS

I. MATRIZES COM AS VARIAÇÕES EM FREQUÊNCIA DO NÚMERO DE CONDIÇÃO E SUAS REPRESENTAÇÕES GRÁFICAS

As imagens contidas nesse anexo mostram as matrizes de variação em frequência do número de condição e suas representações gráficas para diferentes combinações de direções.

| | | | | | | | | | |
|------|------|------|------|------|------|------|------|------|------|
| 1.47 | 1.47 | 1.47 | 1.46 | 1.43 | 1.38 | 1.41 | 1.62 | 1.88 | 2.00 |
| 1.47 | 1.47 | 1.47 | 1.46 | 1.43 | 1.38 | 1.41 | 1.62 | 1.88 | 2.00 |
| 1.47 | 1.47 | 1.47 | 1.46 | 1.43 | 1.38 | 1.40 | 1.62 | 1.88 | 2.00 |
| 1.46 | 1.46 | 1.46 | 1.45 | 1.42 | 1.37 | 1.40 | 1.61 | 1.87 | 1.99 |
| 1.43 | 1.43 | 1.43 | 1.42 | 1.40 | 1.34 | 1.37 | 1.58 | 1.83 | 1.95 |
| 1.38 | 1.38 | 1.38 | 1.37 | 1.34 | 1.29 | 1.32 | 1.52 | 1.76 | 1.88 |
| 1.41 | 1.41 | 1.40 | 1.40 | 1.37 | 1.32 | 1.35 | 1.55 | 1.80 | 1.92 |
| 1.62 | 1.62 | 1.62 | 1.61 | 1.58 | 1.52 | 1.55 | 1.79 | 2.08 | 2.21 |
| 1.88 | 1.88 | 1.88 | 1.87 | 1.83 | 1.76 | 1.80 | 2.08 | 2.41 | 2.56 |
| 2.00 | 2.00 | 2.00 | 1.99 | 1.95 | 1.88 | 1.92 | 2.21 | 2.56 | 2.73 |

(a)

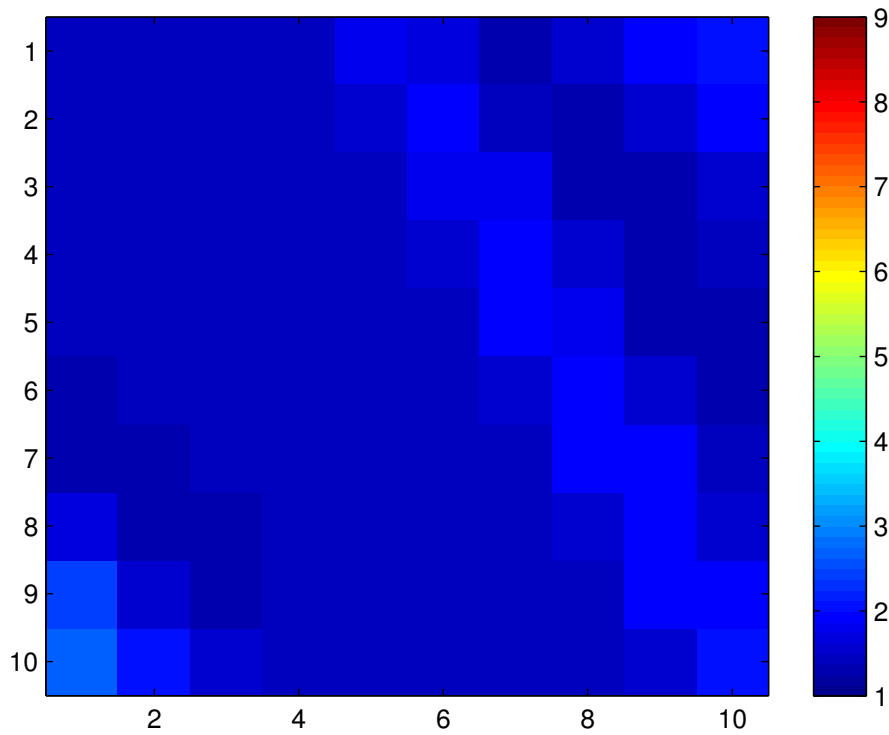


(b)

Figura I.1: Variação do número de condição, $\eta(\mathbf{E}(\omega_1, \omega_2))$, referente à *Wavelet* interpolante (6,6) convencional: $DP1 = 5$, $DU1 = 5$, $DLP2 = 5$, $DLU2 = 5$, $DHP2 = 5$ e $DHU2 = 5$.

| | | | | | | | | | |
|------|------|------|------|------|------|------|------|------|------|
| 1.47 | 1.47 | 1.43 | 1.40 | 1.83 | 1.73 | 1.27 | 1.58 | 1.88 | 2.00 |
| 1.47 | 1.47 | 1.46 | 1.38 | 1.61 | 1.95 | 1.50 | 1.31 | 1.61 | 1.88 |
| 1.47 | 1.47 | 1.47 | 1.43 | 1.40 | 1.87 | 1.83 | 1.30 | 1.37 | 1.62 |
| 1.45 | 1.47 | 1.47 | 1.46 | 1.38 | 1.62 | 1.99 | 1.58 | 1.29 | 1.40 |
| 1.40 | 1.46 | 1.47 | 1.47 | 1.43 | 1.41 | 1.88 | 1.87 | 1.37 | 1.34 |
| 1.29 | 1.42 | 1.46 | 1.47 | 1.46 | 1.38 | 1.62 | 2.00 | 1.61 | 1.34 |
| 1.27 | 1.34 | 1.43 | 1.46 | 1.47 | 1.43 | 1.41 | 1.88 | 1.88 | 1.40 |
| 1.73 | 1.31 | 1.37 | 1.43 | 1.46 | 1.46 | 1.38 | 1.62 | 2.00 | 1.62 |
| 2.38 | 1.52 | 1.37 | 1.38 | 1.43 | 1.46 | 1.43 | 1.41 | 1.88 | 1.88 |
| 2.73 | 2.06 | 1.51 | 1.40 | 1.38 | 1.43 | 1.45 | 1.38 | 1.62 | 2.00 |

(a)

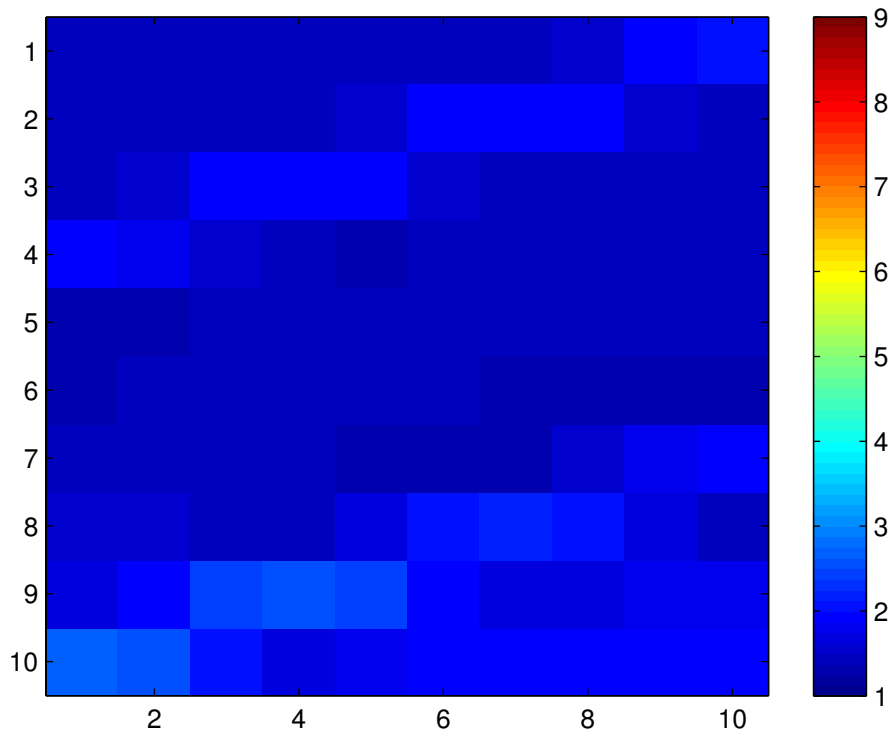


(b)

Figura I.2: Variação do número de condição, $\eta(\mathbf{E}(\omega_1, \omega_2))$, referente à combinação de direções $DP1 = 3$, $DU1 = 3$, $DLP2 = 3$, $DLU2 = 3$, $DHP2 = 3$ e $DHU2 = 3$.

| | | | | | | | | | |
|------|------|------|------|------|------|------|------|------|------|
| 1.47 | 1.47 | 1.47 | 1.46 | 1.43 | 1.38 | 1.41 | 1.62 | 1.88 | 2.00 |
| 1.46 | 1.43 | 1.38 | 1.41 | 1.62 | 1.88 | 2.00 | 1.88 | 1.62 | 1.41 |
| 1.40 | 1.62 | 1.88 | 2.00 | 1.88 | 1.62 | 1.40 | 1.38 | 1.43 | 1.46 |
| 1.99 | 1.87 | 1.61 | 1.40 | 1.37 | 1.42 | 1.45 | 1.46 | 1.46 | 1.46 |
| 1.37 | 1.34 | 1.40 | 1.42 | 1.43 | 1.43 | 1.43 | 1.43 | 1.43 | 1.42 |
| 1.37 | 1.38 | 1.38 | 1.38 | 1.38 | 1.38 | 1.37 | 1.34 | 1.29 | 1.31 |
| 1.41 | 1.41 | 1.40 | 1.40 | 1.37 | 1.30 | 1.27 | 1.52 | 1.79 | 1.92 |
| 1.61 | 1.58 | 1.50 | 1.43 | 1.73 | 2.06 | 2.21 | 2.06 | 1.73 | 1.43 |
| 1.64 | 1.99 | 2.38 | 2.56 | 2.38 | 1.99 | 1.64 | 1.73 | 1.83 | 1.87 |
| 2.73 | 2.53 | 2.11 | 1.73 | 1.84 | 1.95 | 1.99 | 2.00 | 2.00 | 2.00 |

(a)

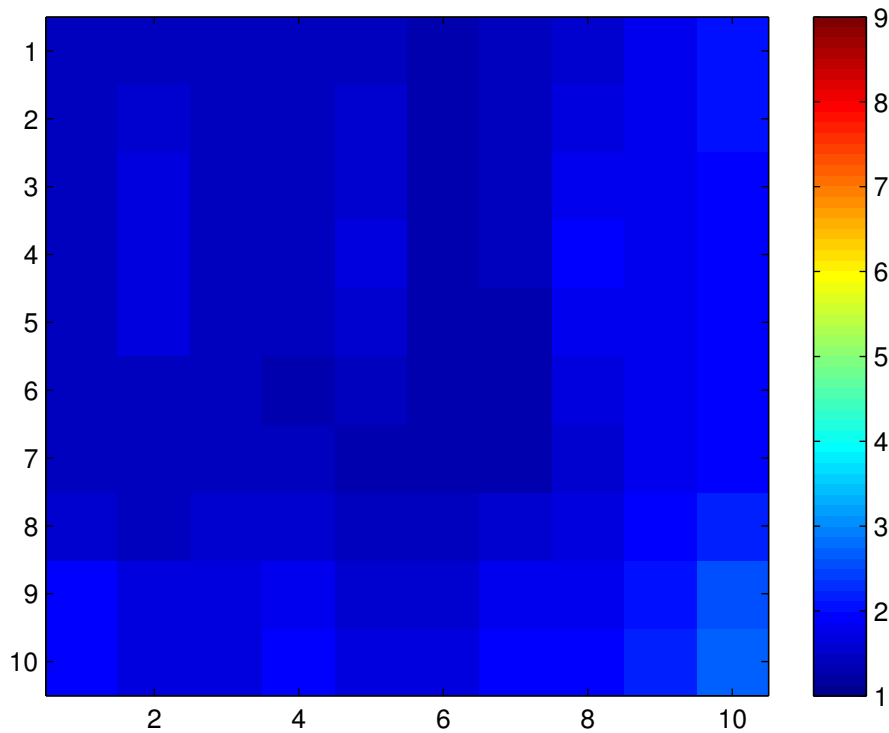


(b)

Figura I.3: Variação do número de condição, $\eta(\mathbf{E}(\omega_1, \omega_2))$, referente à combinação de direções $DP1 = 9$, $DU1 = 9$, $DLP2 = 5$, $DLU2 = 5$, $DHP2 = 5$ e $DHU2 = 5$.

| | | | | | | | | | |
|------|------|------|------|------|------|------|------|------|------|
| 1.47 | 1.35 | 4.39 | 5.14 | 4.39 | 1.27 | 1.41 | 1.48 | 4.48 | 5.14 |
| 1.47 | 1.49 | 4.82 | 5.14 | 3.74 | 1.32 | 1.41 | 1.60 | 4.91 | 5.14 |
| 1.47 | 1.70 | 5.04 | 5.14 | 2.96 | 1.36 | 1.40 | 1.81 | 5.13 | 5.14 |
| 1.46 | 2.21 | 5.10 | 5.11 | 2.21 | 1.37 | 1.40 | 2.32 | 5.20 | 5.10 |
| 1.43 | 2.92 | 5.05 | 4.95 | 1.63 | 1.36 | 1.37 | 3.03 | 5.14 | 4.95 |
| 1.38 | 3.55 | 4.86 | 4.56 | 1.39 | 1.35 | 1.32 | 3.67 | 4.95 | 4.56 |
| 1.41 | 3.84 | 4.47 | 3.84 | 1.37 | 1.40 | 1.35 | 3.98 | 4.57 | 3.84 |
| 1.62 | 3.65 | 3.87 | 2.90 | 1.64 | 1.62 | 1.55 | 3.81 | 4.03 | 3.25 |
| 1.88 | 3.10 | 3.16 | 2.15 | 2.01 | 1.98 | 1.80 | 3.31 | 3.53 | 2.94 |
| 2.00 | 2.45 | 2.45 | 1.99 | 2.46 | 2.49 | 1.92 | 2.77 | 3.13 | 2.73 |

(a)

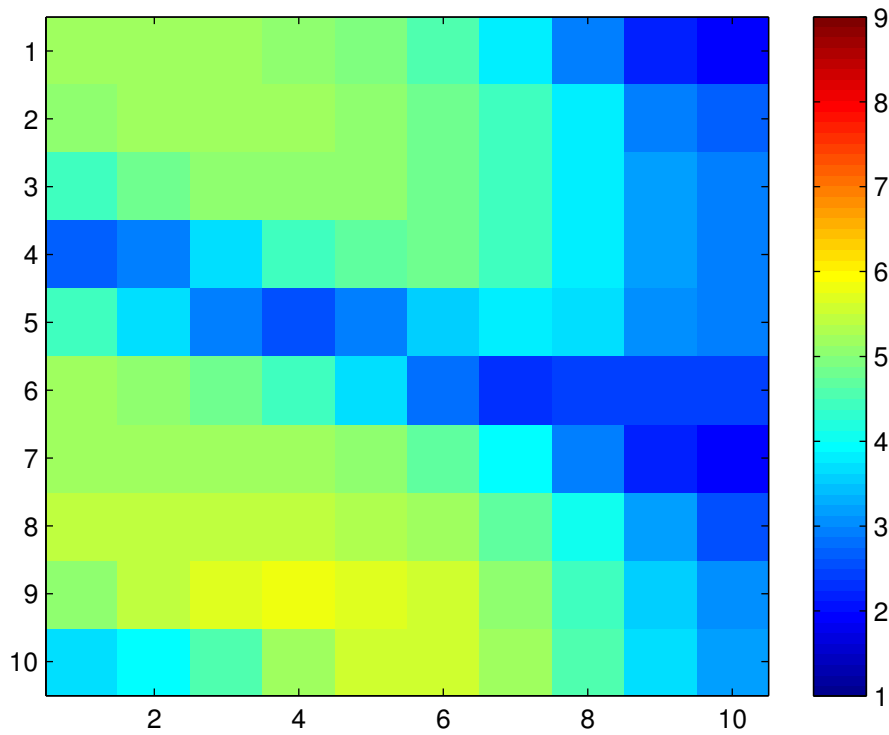


(b)

Figura I.4: Variação do número de condição, $\eta(\mathbf{E}(\omega_1, \omega_2))$, referente à combinação de direções $DP1 = 5$, $DU1 = 5$, $DLP2 = 9$, $DLU2 = 9$, $DHP2 = 5$ e $DHU2 = 5$.

| | | | | | | | | | |
|------|------|------|------|------|------|------|------|------|------|
| 5.14 | 5.14 | 5.14 | 5.10 | 4.95 | 4.56 | 3.84 | 2.90 | 2.16 | 2.00 |
| 5.12 | 5.14 | 5.14 | 5.13 | 5.05 | 4.85 | 4.45 | 3.80 | 2.99 | 2.64 |
| 4.39 | 4.82 | 5.04 | 5.10 | 5.05 | 4.86 | 4.47 | 3.87 | 3.16 | 2.96 |
| 2.67 | 2.96 | 3.74 | 4.38 | 4.74 | 4.76 | 4.45 | 3.87 | 3.16 | 2.96 |
| 4.39 | 3.74 | 2.96 | 2.61 | 2.92 | 3.55 | 3.84 | 3.65 | 3.10 | 2.89 |
| 5.14 | 5.05 | 4.83 | 4.39 | 3.69 | 2.85 | 2.30 | 2.39 | 2.44 | 2.45 |
| 5.22 | 5.22 | 5.22 | 5.18 | 5.03 | 4.63 | 3.90 | 2.97 | 2.16 | 1.92 |
| 5.43 | 5.45 | 5.45 | 5.43 | 5.35 | 5.14 | 4.71 | 4.03 | 3.19 | 2.57 |
| 5.05 | 5.50 | 5.73 | 5.80 | 5.73 | 5.51 | 5.06 | 4.39 | 3.60 | 3.04 |
| 3.66 | 3.96 | 4.61 | 5.24 | 5.61 | 5.61 | 5.23 | 4.56 | 3.74 | 3.14 |

(a)

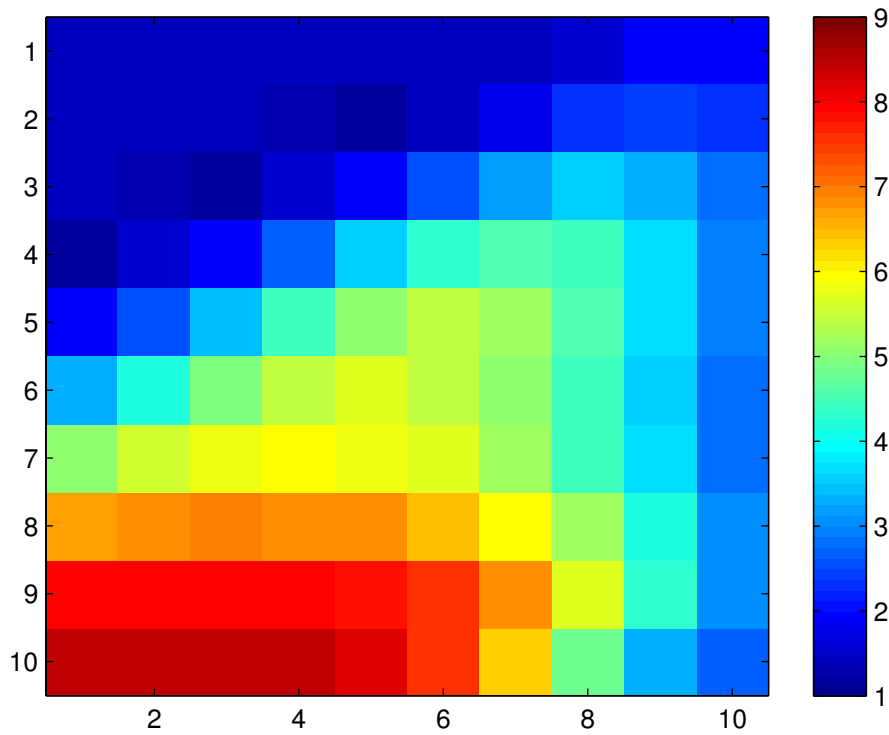


(b)

Figura I.5: Variação do número de condição, $\eta(\mathbf{E}(\omega_1, \omega_2))$, referente à combinação de direções $DP1 = 1$, $DU1 = 5$, $DLP2 = 5$, $DLU2 = 5$, $DHP2 = 5$ e $DHU2 = 5$.

| | | | | | | | | | |
|------|------|------|------|------|------|------|------|------|------|
| 1.47 | 1.47 | 1.47 | 1.46 | 1.43 | 1.38 | 1.41 | 1.62 | 1.88 | 2.00 |
| 1.47 | 1.46 | 1.43 | 1.35 | 1.23 | 1.48 | 1.86 | 2.26 | 2.48 | 2.37 |
| 1.43 | 1.35 | 1.24 | 1.51 | 1.96 | 2.58 | 3.20 | 3.51 | 3.34 | 2.83 |
| 1.24 | 1.50 | 1.97 | 2.67 | 3.52 | 4.28 | 4.62 | 4.40 | 3.74 | 2.95 |
| 1.93 | 2.62 | 3.50 | 4.40 | 5.10 | 5.39 | 5.18 | 4.56 | 3.73 | 2.90 |
| 3.37 | 4.25 | 4.99 | 5.46 | 5.63 | 5.50 | 5.07 | 4.40 | 3.59 | 2.79 |
| 5.09 | 5.60 | 5.85 | 5.92 | 5.86 | 5.63 | 5.18 | 4.49 | 3.66 | 2.84 |
| 6.75 | 6.85 | 6.88 | 6.86 | 6.76 | 6.50 | 5.97 | 5.16 | 4.15 | 3.12 |
| 7.98 | 7.98 | 7.98 | 7.96 | 7.84 | 7.51 | 6.80 | 5.67 | 4.29 | 3.04 |
| 8.49 | 8.49 | 8.49 | 8.43 | 8.18 | 7.52 | 6.34 | 4.79 | 3.37 | 2.73 |

(a)

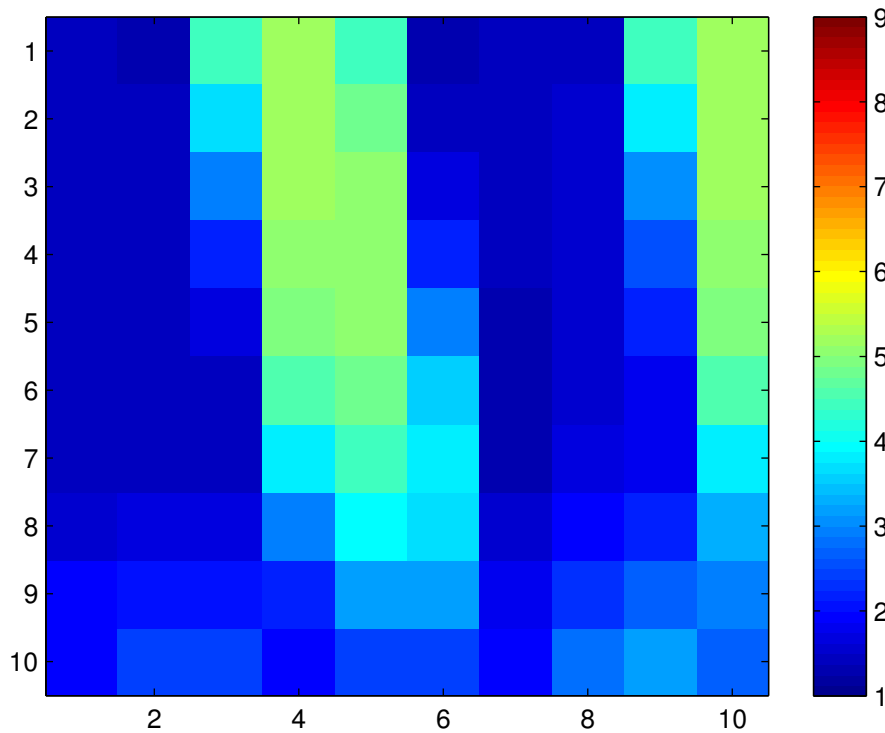


(b)

Figura I.6: Variação do número de condição, $\eta(\mathbf{E}(\omega_1, \omega_2))$, referente à combinação de direções $DP1 = 8$, $DU1 = 5$, $DLP2 = 5$, $DLU2 = 5$, $DHP2 = 5$ e $DHU2 = 5$.

| | | | | | | | | | |
|------|------|------|------|------|------|------|------|------|------|
| 1.47 | 1.35 | 4.39 | 5.14 | 4.39 | 1.27 | 1.41 | 1.48 | 4.48 | 5.14 |
| 1.47 | 1.41 | 3.74 | 5.14 | 4.83 | 1.38 | 1.41 | 1.55 | 3.82 | 5.14 |
| 1.47 | 1.45 | 2.96 | 5.14 | 5.04 | 1.65 | 1.40 | 1.60 | 3.06 | 5.14 |
| 1.46 | 1.46 | 2.21 | 5.11 | 5.11 | 2.23 | 1.40 | 1.61 | 2.51 | 5.10 |
| 1.43 | 1.45 | 1.67 | 4.95 | 5.05 | 2.93 | 1.37 | 1.60 | 2.13 | 4.95 |
| 1.38 | 1.43 | 1.43 | 4.56 | 4.86 | 3.57 | 1.32 | 1.58 | 1.82 | 4.56 |
| 1.41 | 1.49 | 1.40 | 3.84 | 4.47 | 3.86 | 1.35 | 1.65 | 1.80 | 3.84 |
| 1.62 | 1.72 | 1.68 | 2.90 | 3.88 | 3.67 | 1.55 | 1.90 | 2.15 | 3.25 |
| 1.88 | 2.08 | 2.06 | 2.15 | 3.16 | 3.13 | 1.80 | 2.30 | 2.64 | 2.94 |
| 2.00 | 2.45 | 2.45 | 1.99 | 2.46 | 2.49 | 1.92 | 2.77 | 3.13 | 2.73 |

(a)

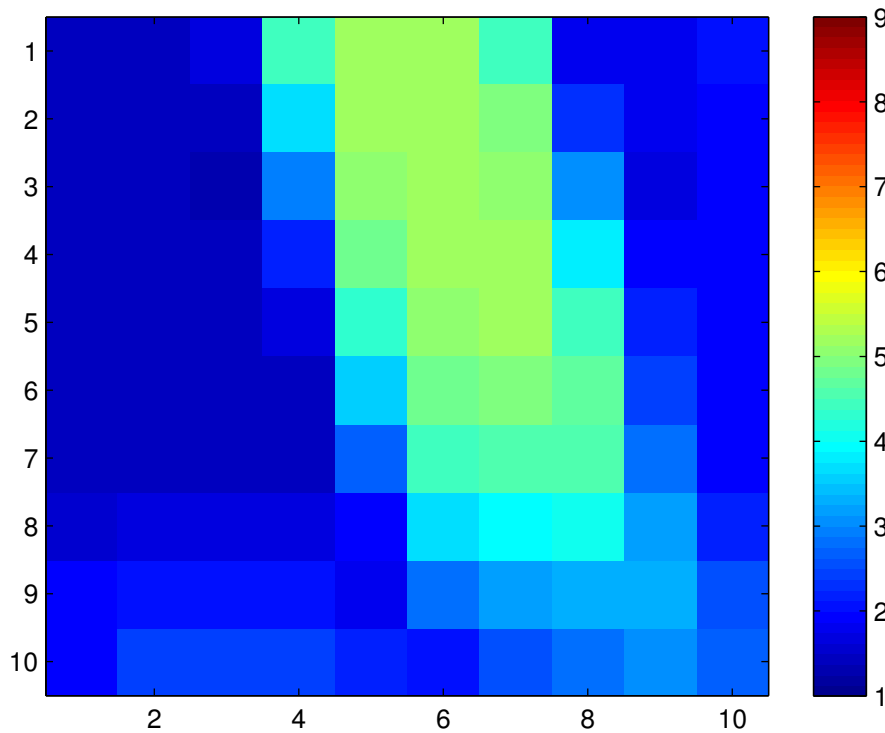


(b)

Figura I.7: Variação do número de condição, $\eta(\mathbf{E}(\omega_1, \omega_2))$, referente à combinação de direções $DP1 = 5$, $DU1 = 5$, $DLP2 = 1$, $DLU2 = 5$, $DHP2 = 5$ e $DHU2 = 5$.

| | | | | | | | | | |
|------|------|------|------|------|------|------|------|------|------|
| 1.47 | 1.45 | 1.70 | 4.39 | 5.14 | 5.16 | 4.46 | 1.82 | 1.86 | 2.00 |
| 1.47 | 1.46 | 1.49 | 3.74 | 5.13 | 5.16 | 4.90 | 2.32 | 1.80 | 2.00 |
| 1.47 | 1.47 | 1.35 | 2.96 | 5.04 | 5.16 | 5.12 | 3.07 | 1.72 | 2.00 |
| 1.46 | 1.46 | 1.40 | 2.21 | 4.81 | 5.15 | 5.18 | 3.84 | 1.89 | 1.99 |
| 1.43 | 1.45 | 1.43 | 1.67 | 4.32 | 5.07 | 5.13 | 4.44 | 2.13 | 1.95 |
| 1.38 | 1.43 | 1.43 | 1.42 | 3.55 | 4.86 | 4.93 | 4.69 | 2.41 | 1.88 |
| 1.41 | 1.49 | 1.49 | 1.40 | 2.65 | 4.40 | 4.55 | 4.53 | 2.84 | 1.92 |
| 1.62 | 1.72 | 1.72 | 1.67 | 1.89 | 3.67 | 3.96 | 4.02 | 3.18 | 2.21 |
| 1.88 | 2.06 | 2.07 | 2.05 | 1.83 | 2.79 | 3.25 | 3.36 | 3.26 | 2.56 |
| 2.00 | 2.38 | 2.46 | 2.45 | 2.13 | 2.07 | 2.56 | 2.78 | 3.06 | 2.73 |

(a)

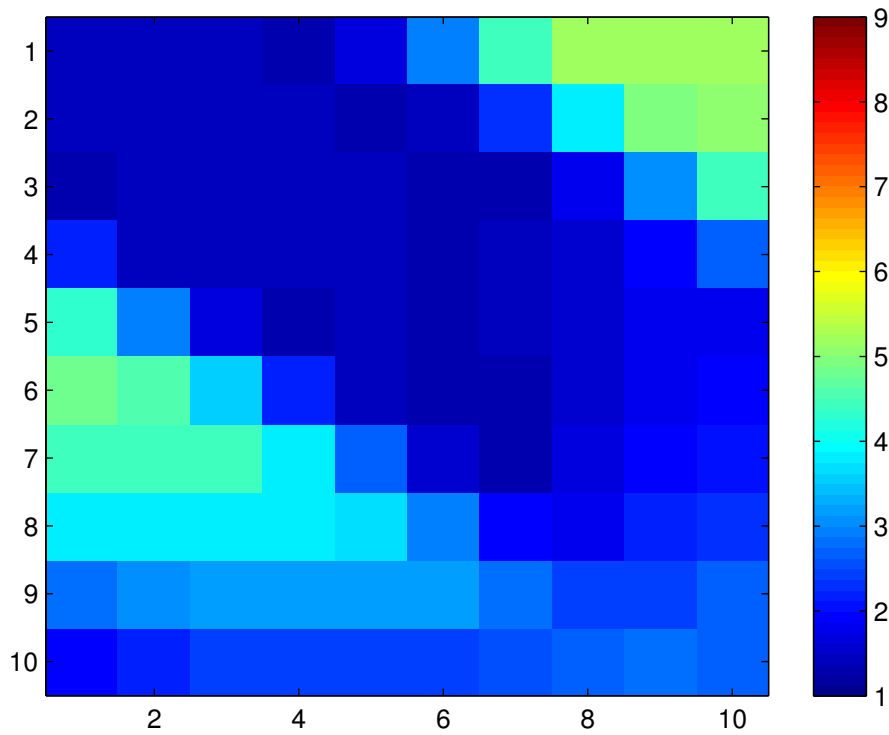


(b)

Figura I.8: Variação do número de condição, $\eta(\mathbf{E}(\omega_1, \omega_2))$, referente à combinação de direções $DP1 = 5$, $DU1 = 5$, $DLP2 = 2$, $DLU2 = 5$, $DHP2 = 5$ e $DHU2 = 5$.

| | | | | | | | | | |
|------|------|------|------|------|------|------|------|------|------|
| 1.47 | 1.47 | 1.45 | 1.34 | 1.66 | 2.98 | 4.46 | 5.18 | 5.23 | 5.14 |
| 1.46 | 1.47 | 1.47 | 1.46 | 1.37 | 1.38 | 2.27 | 3.86 | 4.91 | 5.12 |
| 1.35 | 1.45 | 1.47 | 1.46 | 1.43 | 1.36 | 1.31 | 1.81 | 3.06 | 4.39 |
| 2.21 | 1.48 | 1.40 | 1.45 | 1.43 | 1.37 | 1.40 | 1.55 | 1.89 | 2.70 |
| 4.32 | 2.92 | 1.67 | 1.32 | 1.40 | 1.36 | 1.39 | 1.60 | 1.83 | 1.81 |
| 4.84 | 4.56 | 3.55 | 2.13 | 1.39 | 1.29 | 1.37 | 1.58 | 1.84 | 1.95 |
| 4.47 | 4.46 | 4.38 | 3.84 | 2.65 | 1.57 | 1.35 | 1.63 | 1.91 | 2.03 |
| 3.86 | 3.87 | 3.87 | 3.86 | 3.65 | 2.92 | 1.97 | 1.79 | 2.15 | 2.34 |
| 2.76 | 3.10 | 3.16 | 3.16 | 3.16 | 3.13 | 2.86 | 2.41 | 2.41 | 2.68 |
| 2.00 | 2.18 | 2.38 | 2.45 | 2.47 | 2.50 | 2.56 | 2.68 | 2.79 | 2.73 |

(a)

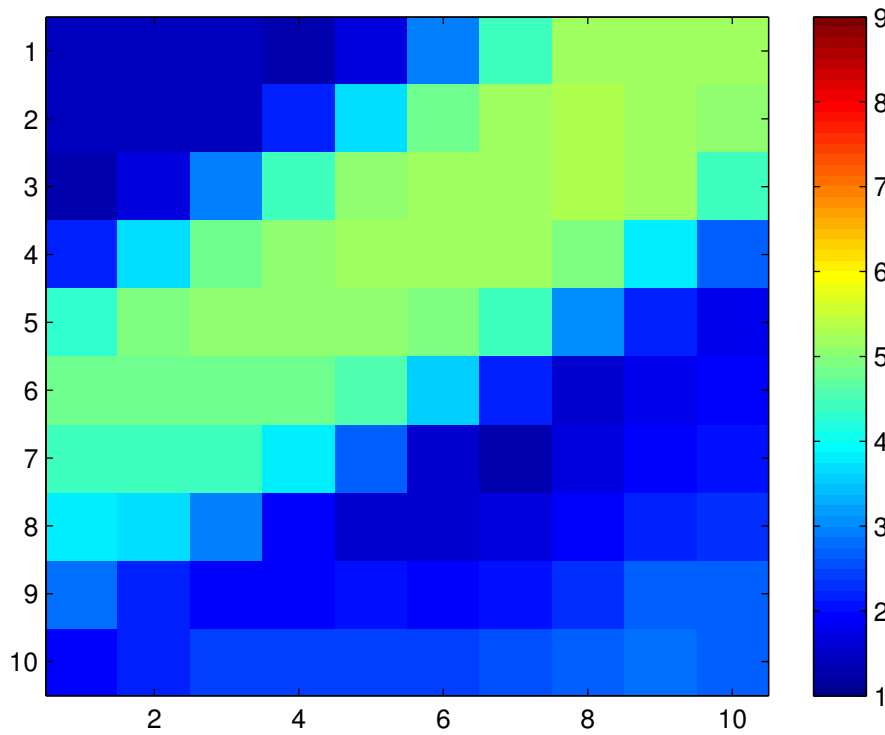


(b)

Figura I.9: Variação do número de condição, $\eta(\mathbf{E}(\omega_1, \omega_2))$, referente à combinação de direções $DP1 = 5$, $DU1 = 5$, $DLP2 = 4$, $DLU2 = 5$, $DHP2 = 5$ e $DHU2 = 5$.

| | | | | | | | | | |
|------|------|------|------|------|------|------|------|------|------|
| 1.47 | 1.47 | 1.45 | 1.34 | 1.66 | 2.98 | 4.46 | 5.18 | 5.23 | 5.14 |
| 1.46 | 1.41 | 1.49 | 2.22 | 3.74 | 4.84 | 5.20 | 5.28 | 5.23 | 5.12 |
| 1.35 | 1.70 | 2.96 | 4.39 | 5.04 | 5.16 | 5.22 | 5.28 | 5.13 | 4.39 |
| 2.21 | 3.73 | 4.81 | 5.11 | 5.13 | 5.15 | 5.18 | 4.94 | 3.81 | 2.70 |
| 4.32 | 4.95 | 5.05 | 5.05 | 5.05 | 4.97 | 4.39 | 3.03 | 2.13 | 1.81 |
| 4.84 | 4.86 | 4.86 | 4.84 | 4.56 | 3.57 | 2.19 | 1.55 | 1.76 | 1.95 |
| 4.47 | 4.46 | 4.38 | 3.84 | 2.65 | 1.57 | 1.35 | 1.63 | 1.91 | 2.03 |
| 3.86 | 3.65 | 2.90 | 1.90 | 1.58 | 1.58 | 1.65 | 1.90 | 2.21 | 2.34 |
| 2.76 | 2.16 | 1.88 | 1.96 | 2.01 | 1.98 | 2.04 | 2.30 | 2.64 | 2.68 |
| 2.00 | 2.18 | 2.38 | 2.45 | 2.47 | 2.50 | 2.56 | 2.68 | 2.79 | 2.73 |

(a)

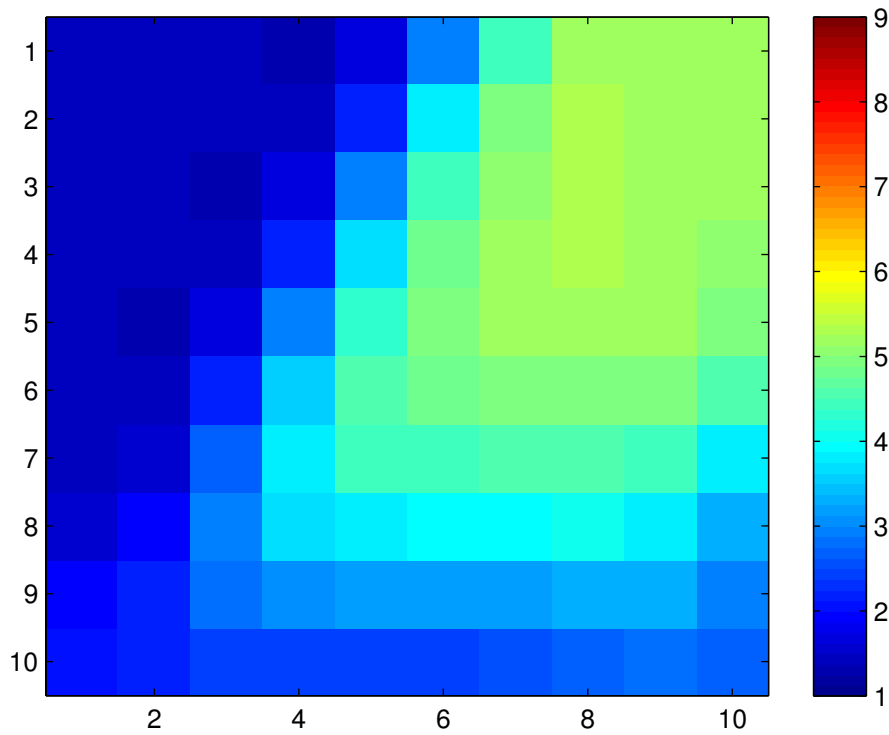


(b)

Figura I.10: Variação do número de condição, $\eta(\mathbf{E}(\omega_1, \omega_2))$, referente à combinação de direções $DP1 = 5$, $DU1 = 5$, $DLP2 = 6$, $DLU2 = 5$, $DHP2 = 5$ e $DHU2 = 5$.

| | | | | | | | | | |
|------|------|------|------|------|------|------|------|------|------|
| 1.47 | 1.47 | 1.45 | 1.34 | 1.66 | 2.98 | 4.46 | 5.18 | 5.23 | 5.14 |
| 1.47 | 1.46 | 1.41 | 1.48 | 2.22 | 3.76 | 4.90 | 5.26 | 5.23 | 5.14 |
| 1.47 | 1.45 | 1.35 | 1.69 | 2.96 | 4.41 | 5.12 | 5.28 | 5.23 | 5.14 |
| 1.46 | 1.40 | 1.48 | 2.21 | 3.73 | 4.83 | 5.18 | 5.26 | 5.22 | 5.10 |
| 1.43 | 1.33 | 1.67 | 2.92 | 4.32 | 4.97 | 5.13 | 5.19 | 5.14 | 4.95 |
| 1.38 | 1.43 | 2.13 | 3.55 | 4.56 | 4.86 | 4.93 | 5.00 | 4.93 | 4.56 |
| 1.41 | 1.62 | 2.64 | 3.84 | 4.38 | 4.49 | 4.55 | 4.61 | 4.49 | 3.84 |
| 1.62 | 1.91 | 2.90 | 3.65 | 3.86 | 3.90 | 3.96 | 4.02 | 3.82 | 3.25 |
| 1.88 | 2.16 | 2.76 | 3.10 | 3.16 | 3.18 | 3.25 | 3.31 | 3.26 | 2.94 |
| 2.00 | 2.18 | 2.38 | 2.45 | 2.47 | 2.50 | 2.56 | 2.68 | 2.79 | 2.73 |

(a)

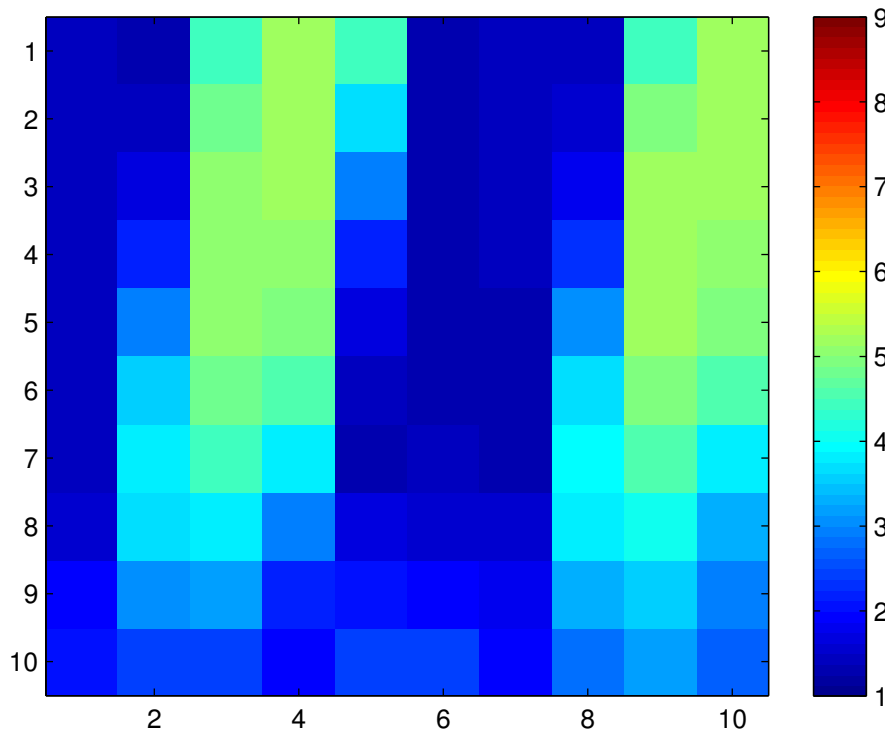


(b)

Figura I.11: Variação do número de condição, $\eta(\mathbf{E}(\omega_1, \omega_2))$, referente à combinação de direções $DP1 = 5$, $DU1 = 5$, $DLP2 = 7$, $DLU2 = 5$, $DHP2 = 5$ e $DHU2 = 5$.

| | | | | | | | | | |
|------|------|------|------|------|------|------|------|------|------|
| 1.47 | 1.35 | 4.39 | 5.14 | 4.39 | 1.27 | 1.41 | 1.48 | 4.48 | 5.14 |
| 1.47 | 1.49 | 4.82 | 5.14 | 3.74 | 1.32 | 1.41 | 1.60 | 4.91 | 5.14 |
| 1.47 | 1.70 | 5.04 | 5.14 | 2.96 | 1.36 | 1.40 | 1.81 | 5.13 | 5.14 |
| 1.46 | 2.21 | 5.10 | 5.11 | 2.21 | 1.37 | 1.40 | 2.32 | 5.20 | 5.10 |
| 1.43 | 2.92 | 5.05 | 4.95 | 1.63 | 1.36 | 1.37 | 3.03 | 5.14 | 4.95 |
| 1.38 | 3.55 | 4.86 | 4.56 | 1.39 | 1.35 | 1.32 | 3.67 | 4.95 | 4.56 |
| 1.41 | 3.84 | 4.47 | 3.84 | 1.37 | 1.40 | 1.35 | 3.98 | 4.57 | 3.84 |
| 1.62 | 3.65 | 3.87 | 2.90 | 1.64 | 1.62 | 1.55 | 3.81 | 4.03 | 3.25 |
| 1.88 | 3.10 | 3.16 | 2.15 | 2.01 | 1.98 | 1.80 | 3.31 | 3.53 | 2.94 |
| 2.00 | 2.45 | 2.45 | 1.99 | 2.46 | 2.49 | 1.92 | 2.77 | 3.13 | 2.73 |

(a)



(b)

Figura I.12: Variação do número de condição, $\eta(\mathbf{E}(\omega_1, \omega_2))$, referente à combinação de direções $DP1 = 5$, $DU1 = 5$, $DLP2 = 9$, $DLU2 = 5$, $DHP2 = 5$ e $DHU2 = 5$.